A FRAMEWORK FOR DEVELOPING EXECUTABLE ARCHITECTURE FOR AERIAL INTELLIGENCE SURVEILLANCE AND RECONNAISSANCE SYSTEMS-OF-SYSTEMS: A SYSTEMS DYNAMICS APPROACH

A Dissertation
Presented to
The Academic Faculty

by

Steven C. Chetcuti

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in the
School of Aerospace Engineering

Georgia Institute of Technology
August 2020

A FRAMEWORK FOR DEVELOPING EXECUTABLE ARCHITECTURE FOR AERIAL INTELLIGENCE SURVEILLANCE AND RECONNAISSANCE SYSTEMS-OF-SYSTEMS: A SYSTEMS DYNAMICS APPROACH

Approved by:

Dr. Dimitri Mavris, Advisor
School of Aerospace Engineering
Georgia Institute of Technology

Dr. Richard Melnyk
Dept. of Civil & Mechanical Engr.
United States Military Academy

Dr. Daniel Schrage
School of Aerospace Engineering
Georgia Institute of Technology

Dr. Brian Wade
TRAC-Monterey
Naval Post-Graduate School

Dr. Michael Steffens
School of Aerospace Engineering
Georgia Institute of Technology

Date Approved: June 25, 2020

To my loving wife Monique and our wonderful boys Luca and Marcus

ACKNOWLEDGEMENTS

Many people have assisted me through this process, to whom I owe a debt of gratitude. I would like to thank my advisor Prof. Dimitri Mavris for his mentorship, guidance, and support throughout this challenging process. I want to thank the members of my committee: Dr. Michael Steffens, Dr. Daniel Schrage, and Dr. (COL) Richard Melnyk for their time, advice, and assistance in this endeavor. Lastly, Dr. (LTC) Brian Wade for his never-ending support and assistance; I could not have done this without him.

Most importantly, I want to thank God for blessing me with these opportunities and for my loving parents, beautiful wife, and awesome little boys. My wife and boys have sacrificed so much so that I could accomplish my goals.

I also want to thank my superiors at West Point: BG (R) Daisie Boettner, PhD; COL Fred Meyer, PhD; COL Brett VanPoppel, PhD; COL Mike Benson, PhD; COL Richard Melnyk, PhD, and LTC Andy Bellochio, PhD. They have provided mentorship, this opportunity, and also allowed me to complete my career goal of commanding a battalion. I also want to thank all the Officers, Warrant Officers, NCOs, Soldiers, and cadets with whom I have served; each has molded me into the officer I am.

I want to thank my mentor and master's advisor from UofM, Prof. Carlos Cesnik for always having faith in me and helping me achieve my goals, wherever they put me.

I also want to thank my Georgia Tech classmates and Research Engineers who helped carry this old man through: Mackenzie Lau, Matt 'Mac' Guckenberger, Dat Huynh, Dr. Alicia Sudol, Anant Girdhar, Martin Morrow, Coby Weit, Chloe Downs, Mike Notham, and Samantha Corrado.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS AND ABBREVIATIONS

A2AD	Anti-Access/Area Denial
ABM	Agent Based Modeling
AISR	Aerial Intelligence, Surveillance, and Reconnaissance
C4ISR	Command, Control, Communications, Computers, Intelligence, Surveillance, and Reconnaissance
CALL	Center for Army Lessons Learned
CNA	Center for Naval Analyses
COA	Course of Action
CTC	Combat Training Center
D3A	Decide, Detect, Deliver, Assess
DA	Defense Acquisitions
DepSecDef	Deputy Secretary of Defense
DES	Discrete Event Simulation
DOD	Department of Defense

DOTmLPF-P	Doctrine, Organization, Training, Materiel, Leadership, Personnel, Facilities, and Policy
FM	Field Manual
FMV	Full-motion Video
Hon.	Honorary
IMINT	Imagery Intelligence
JCIDS	Joint Capability Integration and Development System
JP	Joint Publication
JPP	Joint Planning Process
JSIM	Joint Simulation System
JWARS	Joint Warfare System
LRPF	Long-Range Precision Fires
MBSE	Model-Based Systems Engineering
MDMP	Military Decision-Making Process
MDO	Multi-Domain Operations
MOE	Measure of Effectiveness

MOP	Measure of Performance
M&S	Modeling and Simulation
MSCO	Modeling and Simulation Coordination Office
OPFOR	Opposing Forces
OR	Operations Research
PED	Processing, Exploitation, and Dissemination
PEO	Program Executive Office
PM	Program/Product/Project Manager
PPBD	Planning Programming Budget and Execution
SD	System Dynamics
SIGINT	Signals Intelligence
SME	Subject Matter Expert
SoS	System-of-systems
TC-PED	Tasking, Collection, Processing, Exploitation, and Dissemination
TTP	Tactics, Techniques and Procedures
TXX	Tabletop Exercise

UAS	Unmanned Aerial Systems
V&V	Verification and Validation
VV&A	Verification, Validation, and Accreditation.
VV&T	Verification, Validation, and Testing
WARSIM	Warfighter Simulation 2000

SUMMARY

The world is rapidly evolving as are the future potential combat environments. As the military shifts focus from counter insurgency warfare to near-peer adversaries in a multi-domain operating environment, many modernization efforts will be required. These modernization efforts include repurpose existing systems and systems of systems (SoS) for multi-domain operations, changing the ways by which existing systems are employed, or acquiring new systems.

These modernization efforts must take place across multiple systems and SoS with dozens of different stakeholders, often working on parallel efforts despite the inevitable interaction of the complex SoS in future combat environments. Timelines to develop these systems are in terms of years not decades and are compounded by ever-changing budgetary constraints and geopolitical variations.

The current means to track and document system and SoS developments is the Department of Defense Architecture Framework (DoDAF). DoDAF is a model-based systems engineering framework used to provide visualization of system elements, interfaces, complexities, functions, operational purposes and connections through various viewpoints, models, tables, and graphs in lieu of traditional paper-based accounting methods. For end user and decision-maker orientation, operational viewpoints (such as the OV-1) that are unique to DoDAF are used to provide overall picture and insights into how SoS will be employed and interact within an operating environment.

However, while the DoDAF models are a valuable tool for documenting requirements, the static products are complex and difficult for the average end user to understand and challenging to use for decision-making. In some cases, DoDAF products are treated like a bookkeeping measure and are completed after a decision has already been made. Additionally, the static nature of the architecture fails to demonstrate or determine the efficacy of means or ways trades in against operational measures of performance and effectiveness and little or no guidance exists on when or how to generate executable architecture. Typically, decision makers rely on subject matter expert (SME) estimates to determine requirements and operational impacts. However, SME estimates alone lack scientific backing for requirements development and can prove to be costly if incorrect or not holistic. Alternative means include complex simulations, wargames, and experimental organizations/exercise but these methods are expensive and time consuming. Additionally, simulations that are data in data out lack the learning and interactive benefit.

A need exists for a user-friendly, rapidly developed, computationally inexpensive, and interactive executable architecture (EA) framework that can provide for various means and ways trades to inform strategic-level prioritization and investment decisions. The framework must provide insights into interactions, second-and third-order effects, and feedback and be capable of being developed with partial or missing information. This framework must be adaptable and include elements that are difficult to capture in discrete event type modeling used in the past. It must also serve as a medium to play games, facilitate learning and understanding that are missing in data-driven simulations.

Ongoing research in the field of executable architecture includes the means to be automatically generated from DoDAF products. However, there is no standard format to create DoDAF products. While Systems Modeling Language (SysML™) is the most common means, DoDAF products for existing systems are in a variety of formats such as PowerPoint™, Microsoft Visio™, and Sparx System Enterprise Architect™ making auto-generated executable architecture difficult if not impossible to tailor to end user performance metrics.

A framework for the development of holistic executable architecture for complex systems-of-systems as a method to analyze means and ways trades across the DOTmLP-P (Doctrine, Organization, Training, materiel, Leadership, Personnel, and Policy) spectrum in support of CBA (Capabilities Based Assessment) is proposed. The Aerial Intelligence, Surveillance, and Reconnaissance (AISR) system-of-systems architecture, complete with reachback for the Processing, Exploitation, and Dissemination (PED) of intelligence products, and utilizing a generic Unmanned Aircraft System in support of long-range precision fires as part of a Decide-Detect-Deliver-Assess (D3A) operation was used a case study to experiment and test the framework.

AISR and long-range precision fires (LRPF) has proven to be a critical component of multi-domain warfare. The demand for AISR and rapid development and acquisition of these assets, particularly Unmanned Aerial Systems (UAS) has grown exponentially over the past decade. While research and development focus on UAS, these assets are only part of a complex system-of-systems (SoS) architecture that connects the forward deployed UAS (sensor) to aircraft operators (pilots) who fly the aircraft and intelligence analysts who conduct federated PED of collected data in sanctuary around the globe.

This complex SoS architecture was originally designed and developed with the static DoDAF and with limited use of discrete event simulation type executable models. However, rapid acquisitions to support global requirements over the past decade created an AISR fleet that expanded faster than DoDAF could support. The DOD has outlined a roadmap for future technological improvements to the AISR and PED architecture in addition to additional procurement of UAS over a period from 2017-2042 making it a viable candidate for demonstration of the framework.

Initial application of systems thinking via the use of Systems Tests, Sector Diagrams, Causal Loops, and Model Boundary Chart was utilized reduce system complexity through fundamental understanding of elements, connections, and functions of the SoS. System Dynamics was selected as an appropriate modeling and simulation paradigm using Vensim™ software. Static DoDAF models were mapped to key characteristics of a System Dynamics stock-and-flow model with causal loops to create executable architecture to assess feedback loops and interactions over time.

To evaluate the ability of the executable architecture to evaluate and compare changes to system structure, a baseline PED system model with standardized inputs was recreated from previously published research. The EA was used to demonstrate the ability to evaluate structure, policy, and manning changes to the architecture using Monte Carlo Simulations, Sensitivity Analysis, and Powell optimization. The PED model was then incorporated into the larger executable architecture which included the AISR and D3A fires systems.

Parametric analysis was conducted on the variables of interest. Random distribution functions were applied to replicate the effects of combat and assess the system-of-systems against operational measures of performance and effectiveness. An interactive dashboard environment was created to enable interactive learning for stakeholders via the use of sliders on parametric variables over assumed ranges with instantaneous graphical displays of effects on operational measures of performance and effectiveness over time. This dashboard can be used to conduct manual univariate or multivariate sensitivity It also serves a platform to visualize the effects of various courses of action for a pseudo-tabletop wargaming discussion.

Due to combinatorial and dimensional complexity and stochastic enemy effects, space filling design of experiments with point repetition was applied to sample the input space and accounting for noise variations. This enabled the creation of surrogate models for statistical assessments of complex interactions between the elements of interest against desired operational outputs.

The experiments successfully supported the use of System Dynamics as a means to holistically assess complex systems-of-systems in a rapidly developed, interactive environment that enables trades and could alternatively be used as a gaming and learning tool for stakeholders and decision makers. Findings concerning benefits and limitation are discussed as well as recommendations for future work and improvements.

CHAPTER 1. INTRODUCTION

> "Our Army must regain our overmatch and competitive advantage against emerging threats…we must modernize our capabilities to increase our lethality against emerging regional and global near-peer advisories."
>
> General Mark A. Milley
>
> *Modernization Priorities for the United States Army[1]*

1.1 Persistent and Evolving Global Uncertainty

1.1.1 Setting the Stage

The world and the threats therein are volatile, uncertain, complex, and ambiguous (VUCA), all the while, technology is advancing rapidly. This situation is not unlike that faced by the U.S. Army in the 1970s following the war in Vietnam, as the Army transitioned from draftees and insurgency warfare to a professional army facing potentially quantitatively superior enemies in both the Soviet Union and China [2]. In regards to technical disruption, today's environment bears striking resemblance to the interwar periods between World War I and World War II when new innovations such as radar and sonar coupled with rapid advancements in aircraft, armored vehicles, and aircraft carriers forced strategists and planners to develop new tactics and strategy in the face of massive uncertainty in regards to the effects of these developments and of future war [3].

The current U.S. military has been fixated on counterinsurgency and asymmetrical warfare for nearly 18 years at great financial expense. Meanwhile, the U.S. Army modernization efforts were put on hold for several years as the Army "focused its shrinking budget on maintaining readiness" [4]. During this period, its near-peer adversaries and pacing competitors have studied its doctrine and observed its tactics, all the while, advancing their own militaries in both manning and materiel capabilities [5]. Much like after Vietnam, these requirements have degraded the U.S. Army's technical and tactical proficiency in near-peer combined operations, leaving it woefully unprepared for war not only against sophisticated nations like Russia and China, but also against threats like Iran and North Korea who are benefactors of technological innovations diffused from the larger nations [6]. Innovations like artificial intelligence, unmanned air and land vehicles, autonomy, and machine learning have the potential to revolutionize future battlefields [5]. As a result the U.S. has fallen behind and must modernize rapidly, but modernization requires both acquiring new capabilities and determining how to employ them [1, 3, 7-9].

To compound matters further, the U.S. Army currently has roughly 180,000 troops in 140 countries, which still only satisfies approximately 40-50 percent of the demand from combatant commanders and almost 70 percent of unexpected emergent demands [7]. This means that U.S. military has sunk cost into existing systems and capabilities and despite near-pear threats, has a continued need for these existing systems and capabilities in "hot spots" around the world short-of-full-conflict. Hence. a complete replacement of existing systems or systems-of-systems is unrealistic, and in many cases, unnecessary. Stakeholders must be able to rapidly assess and adjudicate changes (technological, numerical, employment methodology, etc.) to existing systems-of-systems to make them viable in the ongoing and future scenarios.

1.1.2 Third Offset Strategy

On a Joint scale, the Department of Defense has developed the Third Offset Strategy. This strategy, according to former Deputy Secretary of Defense, the Honorary Robert Work aims to “pursue next-generation technology and concepts to assure U.S. Military superiority, but the real focus is strengthening U.S. conventional deterrence” to prevent wars from occurring [8]. The term ‘offset’ is defined as “a consideration or amount that diminishes or balances the effect of a contrary one” [10]. The offset strategies will focus on potential competitors who are reaching parity with the United States in a military area deemed to be critical. This offset plan includes a focus not only on improved technology but on leveraging the option of conduct of war at the operational level (theater/campaign) as a conventional deterrent [8]. Inspired by the Strategic Capabilities Office, Third Offset aims to repurpose previous investment efforts and use them in new innovative ways or integrate individual systems into systems-of-systems.

1.1.3 Multi-Domain Operations

In response to the Third Offset Strategy and the great power competition, the U.S. Army has developed the U.S. Army Multi-Domain Operations (MDO) 2028 concept. This concept provides the skeletal structure of future doctrine and proposes solutions to how the Army, as part of a Joint force, will retain strategic depth, operational advantage and offensive capability necessary for power projection globally as to both deter conflicts and win wars if necessary. The adversaries of the United States will attempt to mitigate its advantages by leveraging layered standoff in all five domains “to separate U.S. forces and our allies in time, space, and function in order to defeat us” [5]. This strategy will require the insertion of “more and more unmanned systems technology…into operational and organizations constructs” to both preserve life and possibly prevent unnecessary escalation in the event of a shoot down [11, 12].

UAS and ISR will play a pivotal role in the Third Offset Strategy and MDO but will be vulnerable to attack in multiple domains. Its employment and architecture must be analyzed.

Figure 1: Key Takeaway

To deter enemies and compete in operations just-short-of-armed-conflict, the MDO concept focuses on the premise that all domains of warfare must be rapidly and continuously integrated. Should deterrence fail and all-out warfare become inevitable, Army elements as part of the Joint Force must:

> **penetrate** and **dis-integrate** enemy anti-access and area denial systems; **exploit** the resulting freedom of maneuver to defeat enemy systems, formations and objectives and to achieve our own strategic objectives; and consolidate gains to force a **return to competition** on terms more favorable to the U.S., our allies and partners [5].

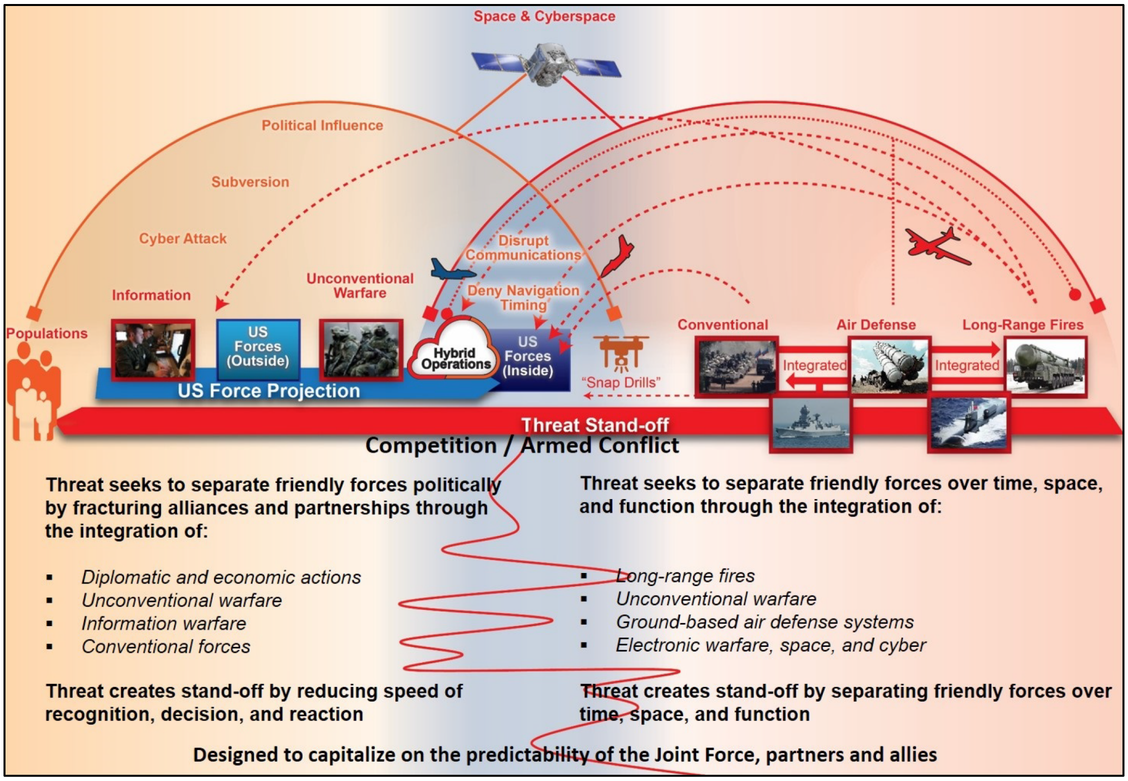


Figure 2: Multi-Domain Operations Concept of Operations from [5]

All future 'challenges' that pose a risk to the Army's ability to match competitors and mitigate standoff in addition to the technological and doctrinal 'opportunities' that allow the U.S. to achieve MDO effects will be evaluated in their ability to 'compete,' 'penetrate,' 'dis-integrate,' 'exploit,' and 're-compete' [13]. The U.S. Army had devised three core tenants to achieve this end: calibrated force posture, multi-domain formations, and convergence. These tenants ensure the U.S. Army, as part of the Joint Force is positioned to maneuver across strategic distances, possess the endurance, capability and capacity to employ effects across all five domains, and converge and integrate these effects rapidly across time and space to defeat the enemy [5]. This new strategy will look across the entire U.S. Army enterprise, to include: doctrine, organization, training, material, leadership and education, personnel, facilities, and policy (DOTmLPF-P) [14]. (See Table 1 [15-17].)

Table 1: Elements of DOTmLPF-P

Doctrine	Fundamental principles by which the military forces or elements thereof guide their actions in support of national objectives. It is authoritative but requires judgment in application.
Organization	The way a unit is organized to accomplish a common mission, cooperate systematically, or provide support to joint warfighting capabilities.
Training	Training includes that of individuals, units and staffs on the necessary doctrine and TTPs necessary to execute their assigned or anticipated missions
Materiel	All items necessary to equip, operate, maintain and support military activities for either combat or administrative purposes (includes ships, tanks, aircraft, repair parts, etc. but not real property/facilities)
Leadership	Influencing people by providing purpose direction and motivation. This element includes the professional education required to prepare leaders to lead the fight
Personnel	The individuals (military or civilian) required to accomplish the assigned mission
Facilities	A real property entity consisting of one or more of the following: a building, a structure, a utility system, pavement, and underlying land
Policy	Any policy issues (DOD, interagency, or international) that may prevent effective implementation of changes to the other seven DOTmLPF-P elemental areas

The Third Offset Strategy and Multi-Domain Operations rely upon both American technological prowess and tactical/strategic ingenuity. Military leaders will have to experiment in order to shape TTPs; doctrine; training; and exercises that allow Joint forces to leverage the technologies developed or perceived as necessary to gain the advantage [6, 8]. If the wargames, models and simulations are creative and rigorous, they can help resolve complex military challenges by reducing the complexity in order to identify the few important factors that constrain friendly forces and systems [3].

With limited resources and competing demands, the military is required to do the most with what it has to accomplish the mission for a given set of requirements, constraints, and assumptions. When unexpected emergent demands arise (contingency operations, natural disasters, civil unrest, etc.) with near-peer conflict possible, it is imperative that Army planners send the right balance of capabilities to support the required mission set given the limited supply and high demand of personnel and materiel elsewhere around the globe.

1.1.4 Future Approach to Acquisitions

The Army established the Army Futures Command (AFC) in 2018 to place all future development, acquisition and testing of future technology under one unified command structure.[13] Within the AFC, the Army established eight cross-functional teams to focus on specific capabilities: Long-range precision fires (LRP), Next generation combat vehicle (NGCV), Future vertical lift (FVL) Network command, control, communication, and intelligence, Assured positioning, navigation, and timing (A-PNT), Air and missile defense (AMD), Soldier lethality (SL), and Synthetic training environment (STE) [18]. These CFTs are focused on the development of new equipment and doctrine necessary to prepare for future high intensity conflict against near-peer threats while simultaneously deterring additional threats and conducting irregular warfare throughout the globe.

As part of the emerging AFC processes, there is a major effort to rework the CBA and JCIDS process from bottom-up process to a top-down Futures Development Process to prioritized work efforts that have the highest payoff for improving the Army's capabilities.

In Task of this emerging process, CFTs must "develop an approach to enable work prioritization that considers challenges, opportunities, and senior leader issues that incorporates outputs from models and simulations, wargames, and experimentation to create the initial rankings" [13]. A rapid method would aid this analysis as much as it would the traditional CBA and JCIDS process; especially if the method is more flexible than traditional data-driven, playback approaches.

1.1.5 Observations

The future of warfare will span all five domains: air, land, sea, cyber and space; much of which is wrought with uncertainty and difficult to quantify. The strategic environment is changing too fast for the Army to dedicate to a single specific operational concept like Air-Land Battle of the 1980s [9]. Myopic focus on a single scenario and strategy has resulted in failures of past acquisition efforts which still hang heavy over the Army. Failed or canceled underperforming programs such as the Future Combat System, the Armed Aerial Scout, the Comanche helicopter, and the distributed Common Ground System-Army cost tax-payers tens of billions of dollars. The Comanche helicopter program alone cost $5.9 billion [7, 9, 19]. Both Congressional and Army leaders have recognized this as problem. Therefore, it is "imperative to combine theory and practice, for doing either in isolation carries the risk of favoring the demands of today over the requirements of tomorrow" [2]. This task will require the ability to visualize and model complex systems and requirements.

Ideally, developers and stakeholders would have unlimited time and monetary resources to develop systems and make informed decisions regarding the acquisition and employment of new capabilities or the modification of existing systems. The real-world, however, is one of bounded rationality: information is limited; the human mind's ability to process information is inadequate; and time is constrained [20]. Therefore, stakeholders require an efficient and holistic means to rapidly visualize systems, foster exploration and conduct informed, traceable, and repeatable means and ways trades for strategic-level financial, structural, and human investment decisions.

The U.S. military must be able to visualize and model complex systems to both acquire new capabilities and determine how to employ them efficiently and effectively.

Figure 3: Key Takeaway

Clearly no method can predict the future, especially in a dynamic environment for which the enemy actions and random occurrences can have significant impact on the results. However, this fact does not diminish the value of assessments and evaluations to provide valuable insights. Current methods, discussed in subsequent chapters rely upon subjective reasoning of subject matter experts, advanced simulations, trial-and-error, and wargames to evaluate a proposed technology's ability to improve desired operational measures of performance and operational measures of effectiveness [21].

What are the limitations and benefits of these methods? Which methods can be used to explore the benefits of integrating systems into systems-of-systems or using them in new and different ways? How can all or some elements of the DOTmLPF-P spectrum be included? Clearly, no single method is enough to solve all problems, and the method utilized depends on the specific problem at hand and the question being evaluated. This leads one to consider alternative methods that are not currently employed to provide rapid assessment, the ability to make trades and play games, and conduct analysis of complex systems that may yield results to confirm subjective assessments or provide insights that might otherwise be overlooked due to the complexity of the problem set or method applied. Subsequent chapters will provide a summary of current methods based on literature review and discussions with subject matter expert practitioners in the field.

1.2 Thesis Format

This chapter introduced the need to develop an efficient, holistically model methods to generate executable architecture for existing complex systems-of-systems to inform generation of a capabilities need statement. A gap was identified in the ability to use simulation to aid both the wargaming and large-scale modeling processes This thesis will use the AISR and PED process to experiment and evaluate doctrine and capabilities; gap-filling technologies; future technology and associated tactics; doctrine and policies to leverage these things against future threats evaluated against operational measures of performance (MOP) and operational measures of effectiveness (MOE).

The second chapter will include a discussion from the initial literature review on the DoD Processes and Activities, the use of Model-based System Engineering (MBSE), the use of Models and Simulation (M&S) in the DoD, and the challenges and gaps associated with these current methods. The chapter will also identify the relevant stakeholders, identify the overall problem, and the overall research objective, along with supporting research questions and hypotheses.

The third chapter of this dissertation will present a more directed literature review on the relevant topics of Systems Thinking methods and M&S paradigms that will be useful in the development of a rapid methodology to support interactive strategic decision-making.

The fourth chapter will be an introduction to the overall framework and approach via the use of a real-world case study for demonstration purposes. It will provide more detail into MBSE used by DoD, specifically DoD Architecture Framework (DoDAF). Using System Thinking methods, key variables and measures of effectiveness and performance are identified. A comparison of M&S paradigms is conducted to the preferred method for a fit-for purposed framework. It will then map the relevant DoDAF models to the creation of the M&S environment used for the evaluation of subsequent research questions. In this chapter the author will also walk through the development and validation of the M&S, its key attributes, and address relevant assumptions used in the creation.

The fifth chapter addresses the second research question focused on the ability of the methodology to be rapidly modified to evaluate structural changes of a SoS.

The six chapter addresses the third research question focused the ability of the methodology to conduct sensitivity analysis to holistically inform the most influential elements of a system-of-systems across the DOTmLPF-P via the use case of AISR PED.

This chapter also demonstrates the method to create surrogate models for rapid exploration and comparison techniques.

The seventh and final chapter summarizes the research problems and questions and provides a summary of the work completed, experiments performed, findings for each of the research questions, conclusions and contributions of the research and identifies area for future work.

CHAPTER 2. PROBLEM FORMULATION

2.1 Existing Methods

2.1.1 DoD Processes and Activities

Following the Cold War, the U.S. Army emphasized a threat-based acquisition paradigm where concepts drove capabilities. In other words, the Army decided how it wanted to fight against a specific threat environment and invested in technology that would support it to that end, rather than having technological developments dictate how it fights [9]. During the War on Terror, however, the Army changed its acquisitions efforts to a capabilities-based paradigm under the Joint Capabilities Integration and Development System (JCIDS) a 'bottom-up' effort generated by Army centers of excellence and end users as part of the larger DoD Processes and Activities shown in Figure 4. Of these overlapping processes and activities, the most tightly interactive are the Defense Acquisition System (DAS), the management process; the Planning Programming Budgeting and Execution (PPBE), the funding process; and JCIDS, the end user requirements process utilizing the Capabilities-Based Assessment (CBA).

These processes are complicated by the asynchronous timelines involved with each. Specifically, the PPBE cycle is calendar driven and is a once-a-year process in four distinct, but interconnected phases that consider the current year, the following (budget) year and the four out-years beyond the budgeting year. JCIDS however, is needs driven and can vary based on the current threat or projected threat (asymmetric versus symmetric; counter-terror versus conventional warfare) based on international politics and world order. Lastly, the DAS is event-driven and requires data, testing, and design to be ready for specific milestones that require Milestone Decision Authority approval to advance to each of its five phases [22].

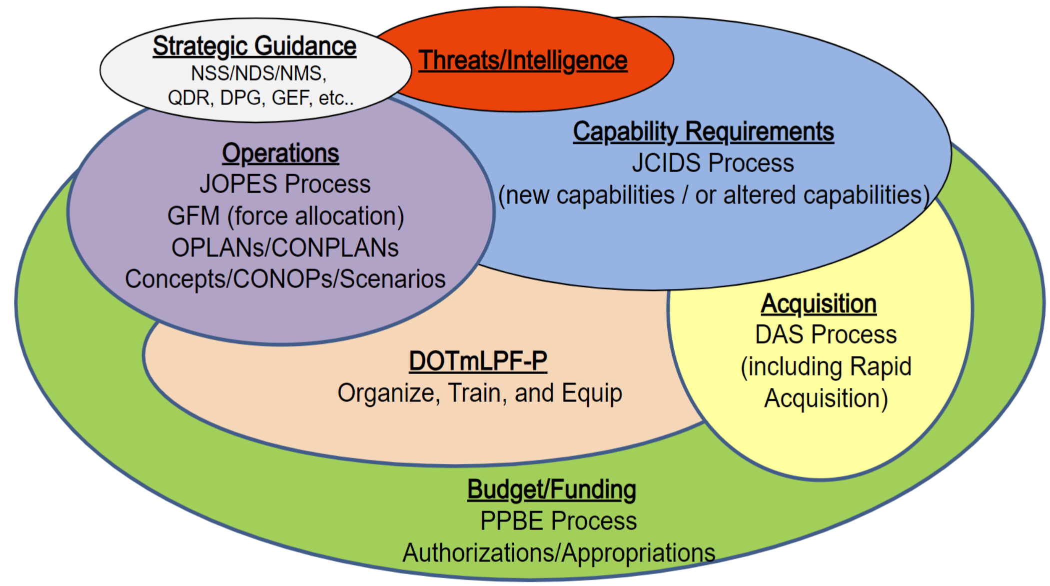


Figure 4: DOD Processes and Activities Interactions from [23]

Trying to synchronize these efforts and timelines in conjunction with DOTmLPF-P, changing strategic guidance, dynamic threats, and ongoing global operations is extremely complicated. Force by bounded rationality, short-sighted decisions may result in capabilities that provide little or no operational benefit for the given investment or, if prolonged in a quagmire of bureaucracy, are no longer relevant or needed when finally fielded.

Furthermore, while Capabilities-Based Assessments are intended to identify and rank the most dangerous gaps, they do so at the lower-level under the assumption that bridging lower-level critical gaps will affect the strategic result end state. However, this often leads to hundreds if not thousands of identified gaps that need to be bridged with technical or tactical solutions. Hence, a need exists for a method of modeling to inform decisions to address lower-level capability gaps with the ability to demonstrate and inform higher level operational outcomes; this method must agile, traceable, and repeatable. Ideally this method would include the ability to evaluate the means, or the capabilities and technologies used to conduct a mission; and the ways; or policies; by which those means are utilized. Additionally, the method should have the ability to incorporate elements from across the DOTmLPF-P spectrum to consider policy changes namely in how it chooses to man, train, organize, and organize the forces under consideration in accordance with the Army's Vision Statement. [24].

2.1.2 Model-Based System Engineering

M&S often plays an important role in military acquisition efforts by helping manage complexity especially given the focus on delivering integrated, network-centric systems-of-systems [25-27]. Because military acquisitions requires collaboration amongst multiple stakeholders, M&S can often serve as an effective means of communication and facilitate shared understanding and insights. In the late 1990's M&S was envisioned under the name of simulation-based acquisitions. In the systems engineering field, this concept has been deemed model-based systems engineering. By using M&S, systems engineers can help manage complexity and interactions at a granular level and then "presenting aggregated impacts and higher-level measures of merit to decision makers" [26]. The International Council on Systems Engineering (INCOSE) defines MBSE as:

Definition: **<u>Model-Based Systems Engineering</u>** is "the formalized application of modeling to support system requirements, design, analysis, verification and validation, beginning in the conceptual design phase and continuing throughout development and later life cycle phases" [28].

MBSE is part of a larger trend towards model-centric approaches rather than the document-centric approaches of the past. In the DoD, this document approach is part of the Joint Capabilities Integration and Development System and the DoD Architecture Format (DoDAF) [26-28].

2.1.3 DoD Architecture Framework (DoDAF)

The rapid development and expansion in use of computers and advanced communications made the battlefield more connected and more complex than ever before. Decision makers in the DoD and in other branches of the government recognized the need to standardize the methods used to develop the critical information systems domain under a joint unified vision to ensure interoperability and cost-effective systems. Created in 1996, it was deemed the Command, Control, Communications, Intelligence Surveillance, and Reconnaissance (C4ISR) Architecture Framework [29]. This goal of this architecture was to describe architectures using multiple views to answer operator questions, support the acquisition efforts, to ensure interoperability. It includes four architecture views (all, operational, system architecture and technical (which included functional, and physical)) executed in three phases (analysis, synthesis, and evaluation) as a means to map operational concepts to measures of performance (MOP) and measures of effectiveness (MOE) for the system (Figure 5). It is important to note that these MOP and MOE used to evaluate the system were for specified system requirements that were assumed to have a benefit on operational performance, not operational MOP and MOE explicitly [30].

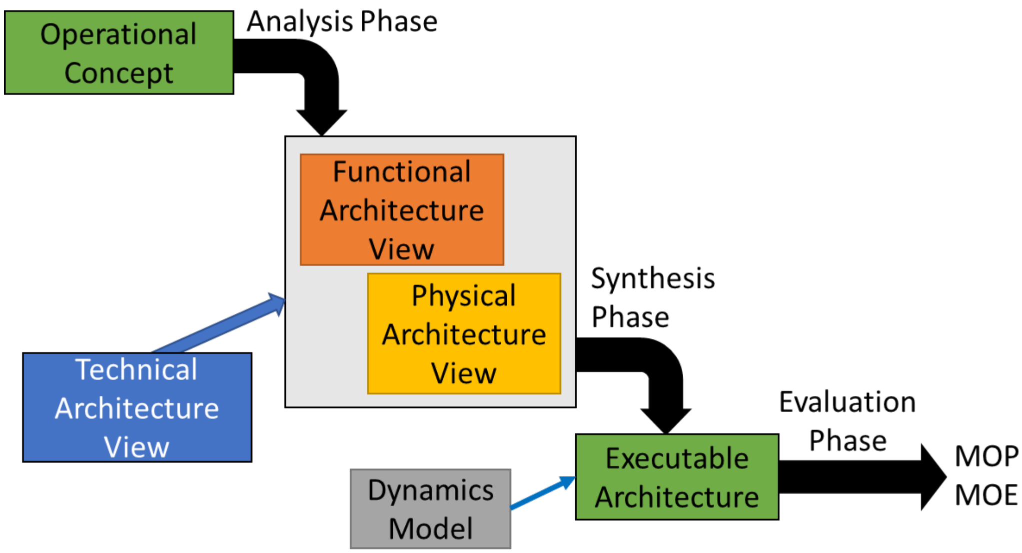


Figure 5: Three Phases of Original C4ISR Architecture Development from [30]

This architecture framework includes a dynamics model to characterize the dynamic behavior of the architecture. These models included state transition diagrams, event traces, and state charts. It is important to note that these were not executable models but static models to depict dynamic behavior of the architecture over time. Executable models were later created to analyze the dynamics of the architecture and identify errors that the static displays of dynamic behavior could not illustrate or detect. At the time, graphical methods and computer simulation methods were employed. Graphical methods included "Colored Petri Nets (Figure 51), Finite State Machines, and Behavior Diagrams" [30].

In February 1998, the DoD mandated the use of C4ISR Architecture Framework on all C4ISR architectures and directed the development of a new architecture to be used on all defense architectures. In 2003, the DoD replaced the C4ISR Architecture Framework with DoDAF Version 1.0 which included additional guidance and was to be applied to all DoD system architectures. The most current version is DoDAF 2.02 introduced in 2010. According to the DoD, the purpose of the DoDAF is to :define concepts and models usable in the DoD's six core processes: JCIDS, PPBE, Defense Acquisitions, Systems Engineering, Operational Planning and Capabilities Portfolio Management [31].

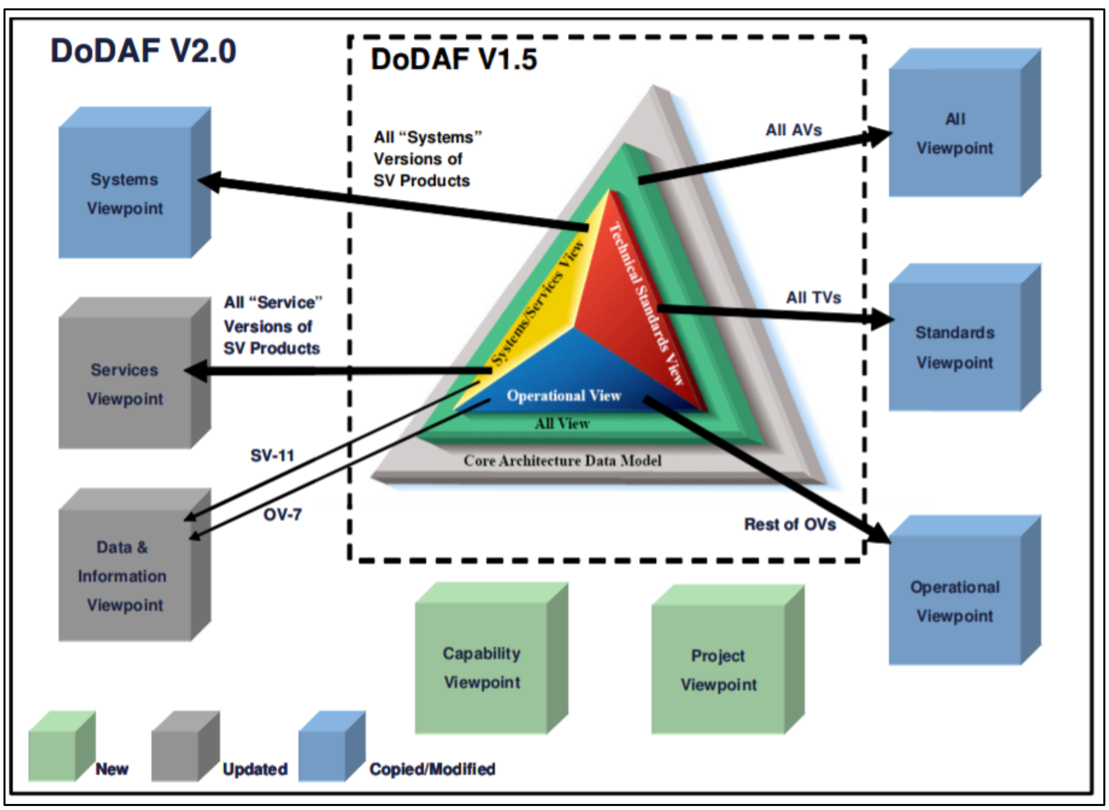


Figure 6: DODAF 2.02 from [31]

DoDAF includes eight types of viewpoints (All, Capability, Data and Information, Operational, Project, Services, Standards, and Systems) totaling 50 different model views [31]. Not every view is required for every project, but each is an available tool to facilitate conversation depending on the use of the architecture.

The primary official means to create these DoDAF products is through the utilization of SysML™, a software modeling tool that was created as a "dialect" of the Unified Modeling Language (UML). This modeling language is meant to provide a means to automatically generate the multitude of interlinked "blueprint" DoDAF products to a very high granularity to ensure consistency in format and detail across all engineering process included in a complete project. In general, the most common and least rigorous usage of SysML™ is to create "pretty pictures" with more advance applications using SysML™ as model simulation to parametrically analyze dynamic behavior with the ability to exercise some behavioral diagrams, including Activity, Sequence, and State Machine diagrams. [32] SysML™ has limited, and rarely used, executable system architecture capability behavioral and parametric specifications able to be simulated, with limited ability to partially execute system interfaces to ensure operation. This execution is done with the help of a simulation engine plug-ins such as MagicDraw Cameo Simulation Toolkit (CST™) or Papyrus Moka™.[33] However, in many cases in the DoD, DoDAF products are often PowerPoint™, Microsoft Visio™, and Sparx System Enterprise Architect™ [34].

2.2 Modeling and Simulation in the Military

2.2.1 Military Modeling Concepts

Military analysts rely on modeling and simulation to approach the innumerable issues and uncertainty that the military faces; from personnel management and training to materiel acquisitions to combat operations. In general, there are three functional areas for military modeling: analysis, training, and acquisition.[35] Major decisions in the military affect human lives, can determine the allocation of hundreds of millions of dollars, and could have global geo-political implications. Modeling and simulation, in some form or another, has existed in the military as long as the military itself; be it in the form of training scenarios with actual force-on-force engagements of foot soldiers, map-based rehearsals, wargaming, mathematical models or advanced computer simulations.

Current DoD simulations fall into three distinct categories: *live*, *virtual*, and *constructive*. As the name implies, *live simulation* involves real systems and real people such as in field exercises. *Virtual simulations* involve using real people to operate simulated systems such as in vehicle and aircraft simulators. *Virtual simulation* also includes incorporating simulated people or systems interacting with real people such as used in large scale theater military exercises to predict results, fabricate enemy movements, etc. Finally, *constructive simulations* are like those discussed in the preceding sections such as mathematical models, advance computer simulations, and wargaming and will be the focus of this discussion and research [36].

Figure 7 from the DoD *Capabilities-Based Assessment (CBA) User's Guide* provides a visualization of the DoD spectrum of analysis approaches. As one moves left on the spectrum, the abstraction, fidelity, and cost decrease while convenience and accessibility increase. As one moves right on the spectrum, the abstraction decreases and operational realism increases, however, so to do costs. It is also important to note that this spectrum is not a menu of options for every system or every problem to be analyzed and addressed. Some problems and systems cannot be analyzed using models either due to modeling limitations or high associated risk due to uncertainty (loss of life, large investment, etc.). Likewise, not all problems or systems may be analyzed using exercises or other live simulation due to complexity or unavailability (future technology, limited resources, deployed operations, etc.).

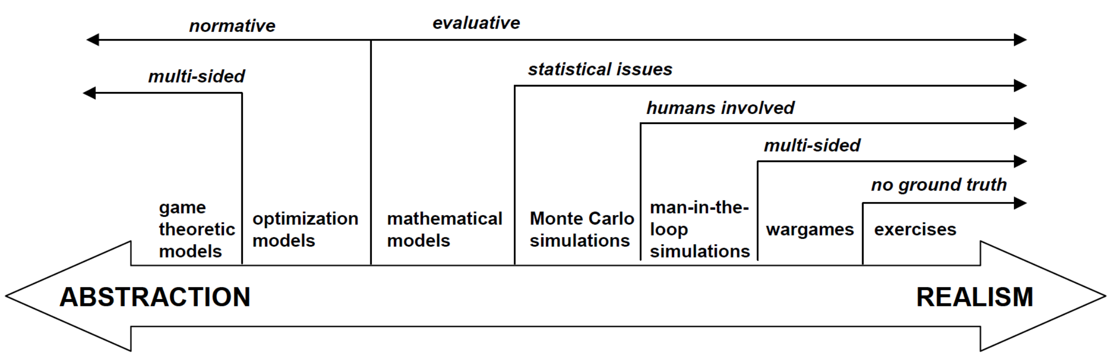


Figure 7: DOD Spectrum of Analysis Approaches from [37]

No method is without its shortcomings. Starting from the left, the normative models are prescribed models to evaluate alternatives to suggest what should be done. Optimization models use rules and mathematical optimization to search a large set of alternatives for local optimums and provide a recommended solution based upon trying to minimize or maximize a given set of quantitative objectives within a given set of assumptions and constraints, hence the abstraction from reality. Game theory models are similar but include the multi-sided competitive nature of combat. The evaluative methods on the right side of the spectrum begin with mathematical models that are used to quantitatively evaluate systems and operations analytically; such as the classic Lanchester Equations and variations thereof.[38] Monte Carlo simulations are an attempt to generate stochastic response by running multiple cases (typically around 10,000) with changing variable inputs to determine the statistical impacts of random events. Man-in-the-loop simulations involve a human player versus computer or machine. Typically, A human on human multi-sided simulation is typically referred to as a wargame, though one-side wargames are one of the many wargame types. Finally, exercises, though more representative of real operations have no ground truth as they are subject to human judgement and unpredictable human decisions leading to lack of repeatability for experiments and statistical issues [37].

All the methods are bounded by their own limitations on what they can and cannot include given a problem, constraints, and assumptions; echoing the "all models are wrong" principle. More realism means more human involvement and bias and unrepeatable uncertainty due to the variability and randomness in human behavior. Additionally, methods involving humans must run in real time making large number of alternatives impossible. For problems requiring the evaluation of tens if not thousands of alternatives be analyzed require increased speed and hence abstraction and modeler bias.

2.2.2 Wargaming

2.2.2.1 Wargaming Defined

The Joint Planning Process vaguely defines wargames as "representations of conflict or competition in a synthetic environment, in which people make decision and respond to the consequences of those decisions."[39] The Army's MDMP handbook also vaguely defines war-gaming as a "conscious attempt to visualize the flow of operations given the friendly force's strengths and disposition, the enemy's capabilities and possible [courses of action] COAs" [40]. Still, *FM 6-0 Command and Staff Organization and Operations*, defines war-gaming as a "disciplined process, with rules and steps that attempt to visualize the flow of the operation" [41]. It goes on to detail eight specific steps and three methods (belt, avenue-in-depth, and box) using either maps of the area of operation or lines of effort. The most basic form of this process is manual and uses a tabletop approach with maps, matrices, and templates, while the more sophisticated methods may use computer-aided modeling and simulation Regardless, a wargame should follow an action, reaction, and counteraction methods force interactions and can be applied to any type of operation. According to the Center for Army Analysis Strategic Wargaming Division, wargaming designer LTC Brian Wade, PhD: "the analytic value of the wargame is more about insights from the players than from the actual outcome of the game. The increased detail required to fully explore the outcome of interactions is usually more appropriate for follow-on computer simulations" [42].

While wargaming has existed in some form or another since centuries before the establishment of the United States and has been in continuous use throughout the existence of the U.S. Army, recent focus (within the past 20 years) has been on campaign analysis, especially regarding modeling and simulation. The difference between wargaming and campaign analysis is subtle but important. Typically what differentiates a wargame from an exercise or a mere simulation is the existence of a thinking opponent and the inclusion of competition, though modern wargaming includes one-sided variants [43]. Campaign analysis will typically use fixed scenarios and varies the numerical and technological components each time it is run, which is good for procurement decision by allowing campaign comparisons and determining the most cost-effective mix of capabilities and forces. Campaign analysis essentially focused on making trades of means with each iteration without altering the ways to quantitatively compare results. By contrast, wargaming focuses on experimenting with the scenario itself including where the conflict occurs, who is participating, and the strategies involved; essentially varying the ways and the means to determine a better way to approach a problem [6]. According to DepSecDef. Work and General Paul Selva:

> Having human players deciding their best actions, given their circumstances and expectations of what their adversary will do in response, is essential to effective wargames. In this, they are distinct from the typical mathematical or campaign modeling that largely seeks to remove dynamic human action from consideration [3].

Wargames in general should have three characteristics; they should focus on war or military activities, there should be some game or competitive aspect, and there should be some form of educational facet. Wargaming can be beneficial in evaluating the effects of difficult to quantify elements such as policy, technology, tactics, and political effects influence the desired result. Fundamentally, a wargame is simply a “tool for exploring and informing human decision-making” [44]. Therefore, the wargame can be supplemented by modeling and simulation, but must be in a way that allows decision-makers and planners to experiment with and visualize the effects of their decisions in the face of uncertainty.

The best wargames duplicate the realm of human judgement in conditions of uncertainty and unpredictability that utilize qualitative analytical techniques to allow for the exposure of idea and insights that are not readily obtainable through typical quantitative modeling, simulation, and optimization techniques**.** Players must be able to decide their actions with incomplete and imperfect knowledge and observe the response feedback to be effective. The games should not be rigged to a specific outcome nor create "self-fulfilling, self-congratulatory, self-deluding, or self-limiting prophecies" [3].

Strategic and high-operational wargames should eliminate non-important complexities while still modeling complex behavior. No two wargames are alike and there is no defined single-standard method for conducting a wargame.

2.2.2.2 Brief History of Wargames

Modern wargaming can trace its roots to the reductionist ideals of the Enlightenment period, the premise of which is that one can describe complex phenomenon by analyzing its fundamental constituent pieces. This thought lead to the precursor of modern wargaming developed by Scotsman John Clerk who used model ships to step through battles and analyze the influence of ship arrangements, ship maneuvers, wind, and naval firepower had on combat effectiveness. However some experts contend that while providing valuable insights, his efforts, due to the lack of a thinking opponent were actually early modeling (representation of reality) and simulation (representation of reality over time) not wargames [43]. Nevertheless, Clerk's method of attempting to solve military problems using linear, deductive mathematics is the precedent of modern, quantitative methods still in use [45].

Lieutenant William McCarty Little who helped found the Naval War College in 1884, introduce wargaming there in 1887 using blackboards, maps, and miniature ships. Wargames have been conducted at the Naval War College annually since. Soon thereafter, wargames had a direct influence on the Navy's budget and helped convince Congress to fun the Cape Cod Canal [43].

Lieutenant Little was an early systems thinker; he saw the value of looking at the whole messy, complicated processes and forced players to consider the broader complex problems holistically rather than to characterize the systems by their individual parts. This holistic systems-thinking was particularly innovating since the Industrial Revolution was underway and reductionist and technology focused solutions were in vogue. In 1899, the Army created its War College and with the help Lieutenant Little, introduce wargaming into its program as well [45].

2.2.3 Evolution of Models and Simulation in the Military

With the rapid increase in technological developments wargaming instead gave way to the field of operations research (OR), which focused on, probability, statistics, numerical optimization and the newly expanding fields of modeling and simulation in its methods. These methods, while valuable, eschew the quantitative features in favor of game theory and other classic Cartesian-Newtonian methods based on reductionism, disciplinarity and multidisciplinarity [45]. The Cartesian-Newtonian scientific method attempts to divide complex problems into different parts which are investigated to gain insight into the behavior as a whole; assuming their behavior is the same when aggregated. The problem that arises is that the element of warfare are interdependent and therefore problematic, "a set of problems in which one problem arises though other problems which then support it" [46]. Like attempting to optimize individual elements at the risk of failing to optimize the whole, solving one problem may make the others worse. However, that is not to say that OR was not beneficial nor successful, as it was able to determine ways to allocated scarce Allied resources in the Battle of the Atlantic in WWII, amongst other successes, which led to the creation of the RAND Corporation in 1948 and OR's continued and heavy use today in all branches of service.

When Robert McNamara took office as the Secretary of Defense under President John F. Kennedy in 1961, the former CEO of General Motors attempted to merge proven OR techniques (namely M&S) with management methods from civilian sector. His aim was to improve DOD acquisition efforts by completing a "life cycle cost analysis to learn what a proposal would really cost and then use OR techniques to estimate military utility" [43]. This method was the precursor to what would later be codified as the DOD Planning Programming, Budgeting and Execution (PPBE) and was used to develop the concept of an air-mobile division in the Army, amongst other innovations.

'Wargaming" of the 1980s and 1990s became increasingly technical and simulation-dependent with a heavy reliance on commercial war-gaming software. These software packages allowed for quicker development of wargames to develop plans without the use of maneuver troops. The models were attrition focused, and while they could predict attrition rates, they were unable to adjudicate the strategic impacts, nor did they allow decision makers to see the complex interactions of their decisions [45].

The Department of Defense spent more money on modeling and simulation "wargaming" in the 1990s than they did in the previous decades combined. However, as noted in his reflective paper "Army Operations Research—Historical Perspectives and Lessons Learned", Dr. Seth Bonder, arguably one of the greatest contributors to Army OR and a Military Operations Research Society Fellow, "millions, if not billions of dollars, have been spent on the development of JSIMS, WARSIM, and JWARS over a number of years without, unfortunately ,any appreciable use" [38]. These, like all of the simulation-dependent Global Wargames of the time "focused on analytical outcomes…rather than on the deliberative processes employed by the players and their adversaries, which are equally important [45]. At the time, RAND identified the idea that "a more comprehensive adjudication of armed conflicts" was required and without such, more computing power would just produce more incorrect answers with better graphics with different software looking at individual regimes an giving different answers to the same questions [43]. This trend of substituting computing power for human effects and system impacts has continued to the present day; leading to DepSecDef Works' concerns that the DOD has lost its ability to conduct effective wargames [3].

2.2.4 Modern Military M&S Hierarchy

The Defense Modeling and Simulation Coordination Office maintains a catalogue of thousands of previously produced models designed and used for a variety of problem sets, though most are not maintained and require specific used in their development [47]. Many of the models and simulations were designed as specific acquisition tools or for operations research. These models may range from large-scale combat modeling and simulation to estimate outcomes to specific engineering performance models. Other modeling and simulation methodologies used include war game simulations, agent-based simulations, stochastic, models, linear programming, and other tools from the realm of operations research [48].

Figure 8 depicts the DOD modeling hierarchy in terms of warfighting scope. The base of the hierarchy is engineering models with very high resolution and very low aggregation. These are modeling physical systems and system characteristics such a single radar system or missile and great care must be taken to include detailed, accurate physics.

These are usually a snapshot in time or focus on the order of only a few minutes of operation. Engagement level models would be a force-on-force scenario of only a few numbers of systems or small units. While engagement level models will still include relevant physics, they are primarily concerned with tactics of the engagement and determining strengths and weaknesses against enemy capabilities.

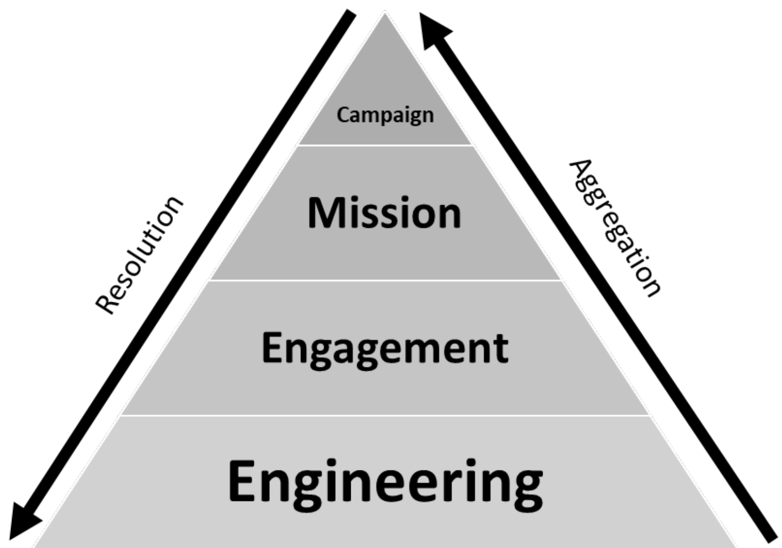


Figure 8: DOD Modeling Hierarchy from [35]

These engagement simulations depict only a few minutes [35] to a few hours. Mission (or Raid) level simulations would contain larger force-on-force simulations for a longer period of simulated time (hours or a few days). Physics are limited to only key contributors at lower fidelity, usually through the employment of data generated by models lower in the hierarchy if they exist. Campaign analysis is the highest level of abstraction and lowest resolution meant to depict an abstract representation of an entire war on the order of weeks to months [37].

In the past few years, the U.S. Air Force has invested in a new “framework” rather than a single model to join the existing Air Force Standard Analysis Toolkit (AFSAT). This framework, the Analytic Framework for Simulation (AFSIM), is an agent-based simulation framework developed by Boeing and now managed by the Air Force Research Laboratory [35]. AFSIM is used by over 150 organizations to include the Army. However, AFSIM, while a framework for capable of rapid scenario composition is currently limited to engineering through mission level simulation and lacks the capability to model a campaign needed to incorporate every aspect (logistics, communications, space, cyber, intangibles, etc.) over a series of related major operations. To do so would not only be an overwhelming task in an agent-based simulation, but would be costly monetarily and computationally, requiring intensive computation power and time for multiple (if not thousands) iterations to run [49].

2.2.5 Modern M&S Challenges

The *Modeling and Simulation Guidance for The Acquisition Workforce* emphasizes that the current M&S methods can “model many things quite credibly today, such as physical capabilities, natural phenomena, and physics-based interactions, it is much more difficult to reliably represent things we understand less well, such as human behavior, reliability, and emergent behaviors of complex systems” [26].

As will be expounded upon in Section 3.3.2, Discrete Events Simulation (DES) allows users to track events through time and explore progress through the system but use a fixed structure to do so. While it is good for analyzing processes and can include stochastics, it is primarily concerned with process details and does not overtly seek to analyze feedback loops or emergent behavior and has difficulty capturing and quantifying the effects of human behavior.

Agent-based modeling (expounded upon in Section 3.3.3) can be constructed without *a priori* knowledge regarding global interdependencies and given a set of preprogrammed rules, can inform emergent complex dynamic behavior at the global level. However, these agents are brittle and will fail when presented with unusual data or unpredictable human behavior. Additionally, these types of models must be heavily aggregated at each level to reduce computational expense. Changes must be made to agent behavior and the entire model run again, limiting the number of means and ways trades that can be made. Additionally, like the Rand publication suggests, ABM tend to only recreate a playback of a scenario and are not an interactive simulation. The lack of interaction results in a loss of learning by the players and insights gained from the players that traditional wargaming provides.

All three commonly used military M&S paradigms are challenged with the ability to include difficult to quantify variable such as IO, space, and cyber effects. However, inclusion of such factors is a current topic of research with some approaches such as the Implicit Model Development Process augmentation to the Model Development Process have been proposed by researchers at the Naval Post Graduate School Operations Research department [50].

2.3 Defining the Problem

2.3.1 Stakeholders

There are many stakeholders that can benefit from having a rapid means to execute and analyze complex system-of-systems architecture. Ideally in a holistic systems-thinking manner can provide a balance between purely manual methods and fully virtual simulation methods. There is not currently an overarching model for analysis of this entire system to provide a common architecture and foundation for communication between all these entities.

A sample lists of stakeholders includes Army operational commanders and planners at every echelon from strategic commanders down to brigades responsible for conduct of operations (STRATCOM FORSCOM, COCOMs, INSCOM), doctrine developers and training organizations (TRADOC, TCM), acquisition and modernization organizations (AMC, FUTURES, PEOs), original equipment manufacturers, industry vendors, Operations Research personnel, Congress, and senior leaders (DOD, JCS, Service).

Table 2: Stakeholders Benefit from Rapidly Developed Executable Architecture

Stakeholder Group	Needs and Uses
Developers	Generate future capabilities Improve existing capabilities Develop doctrine, strategy, TTPs
Researchers	Conduct Analysis Predict operational metrics
Operational Commanders and Planners	Develop plans Conducted operations and PED Develop strategy and TTPs
Decision Makers	Balance system effectiveness against affordability and risk Allocate funds

2.3.1.1 Developers

Developers include developers of technology, training, and doctrine. In the realm of technology development, this group includes those involved in research and development (R&D) of new technologies, Army Futures Command (FUTURES) who generates requirements, Army Acquisitions personnel who must assess technological and material alternatives to satisfy capability needs and gaps, and the original equipment manufacturers and vendors that develop and supply the material solutions.

A complex system-of-systems architecture has a multitude of stakeholders in the acquisition realm alone. For example, air vehicles themselves are designed, delivered and sustained by PM UAS under PEO Aviation, while sensor systems and data processing systems fall under PEO Intelligence, Electronic Warfare, and Sensors (IEW&S) and are managed by one of the PMs within that organization (PM Aircraft Survivability Equipment (ASE); PM-DCGS-A; PM Sensors Aerial Intelligence (PM-SAI); PM Tactical Exploitation National Capabilities-Army (PM-TENCAP), or PM Electronic Warfare & Cyber (PM-EWC). TRADOC Capabilities Managers (TCMs) act as the centralized manager for all DOTmLPF-P solutions for the Army to ensure their integration into the overall employment strategy for the Army. They serve as the liaison between operational commands who will use the materiel solutions and the developers, testers and PMs responsible for the acquisition of said equipment. As one would expect, there are multiple TCMs involved in this complete AISR architecture, to include TCM Intel and Sensor, TCM Senor Processing, TCM UAS subordinate to the Intelligence and Aviation Centers of Excellence.

These entities must be able to evaluate the combat effectiveness of current capabilities against existing doctrine and against future threats; in absence of actual technology or material solutions being available these assessments are conducted using a variety of modeling and simulation. However, the most common means of modeling and simulation allow for technology trades to compare effects on capabilities related to customer requirements but not in operational environments. Other modeling and simulation allow for comparison of material solutions in a fixed environment that allows means trades but often has fixed ways of employment and do not include difficult to quantify factors to reduce noise for comparison.

2.3.1.2 Researchers

Researchers includes Army Operations Research personnel whose primary responsibility is to conduct studies and analysis to inform acquisitions and decision-making processes. This community informs acquisitions, manpower, modernization, and strategic decisions using numerical optimization, data analysis, machine learning, and modeling and simulation. It is this responsibility of this community to develop, maintain, and run the primary simulations in use today at every level (engineering, engagement, mission/battle, and theatre/campaign). The community uses advance models that can me massively time consuming and computationally expensive to run for each iteration. They are currently looking for ways to better conduct campaign level analysis that allows for means and ways trades that can incorporate difficult to quantify factors (cyber, space, training, etc.) could benefit from a methodology that allows initial analysis at a high level of aggregation and lower complexity that can aid in the identification of the most influential factors and validate assumption that are used in their high fidelity computational expensive models and wargames. Theoretically, this will reduce computing time and expense and improve estimates that are used for the larger combat models rather than estimates of ISR effectiveness.

2.3.1.3 <u>Operational Commanders and Planners</u>

Operational Commanders and Planners extends from joint Combatant Commands (COCOMs) and Army Service Component Commands (ASCC) down to battalion level commands conducting tactical-level operations (flying aircraft, conducting PED, and coordinating fires). These commands develop plans for all noncombat and combat operations utilizing the Military Decision-Making Process (MDMP) to inform the Army Design Process and aid the commander in making and executing decisions. This detailed process includes the generation of multiple courses of action (COAs) and the analysis of these COAs via wargaming (either by lines of effort or map-based maneuver), to not only select the best COA but create decision points with branches and sequels to allow commanders to adapt plans and mitigate risks to the mission with changing conditions and enemy actions.

The results of this analysis should better inform total assets combination required, the critical shortfalls in the existing system, identify vulnerabilities, and inform changes to the architecture routing in the development of new doctrine of the MDTF. Commanders are currently limited by the existing system and manual MDMP that are either reactionary or limited by the human inability to recognize non-linear causality that is typical of a dynamically complex problem and could benefit from systems thinking and M&S tools to make complex trades and improve their organizational standards.

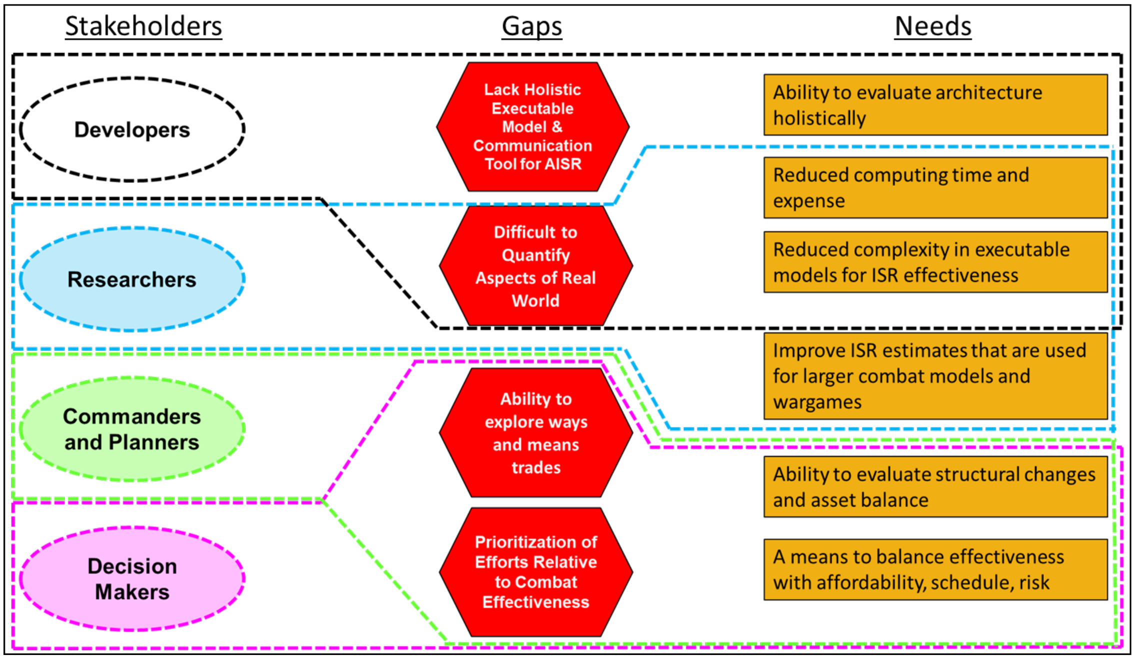


Figure 9: Overlapping Stakeholder Gaps and Needs

2.3.1.4 Decision Makers

Decision makers are senior civilian including the Secretary of Defense, Service Secretaries, and the Joint Chiefs of Staff, and the respective service component department level staffs. These senior leaders in the Department of Defense enact policy for the Joint force to accomplish strategic and transformative goals and ensure that the overarching U.S. strategic guidance created by the president and funded by congress is supported throughout each of the DOTmLPF-P domains within each branch of service. These senior defense leaders liaise with Congress to obtain funding for training and acquisition programs. They are tasked with balancing capabilities given finite resources while ensuring the U.S. military dominance in warfare and influence in peacetime global contingency operations. These decision makers must set requirements to balance system effectiveness against affordability, schedule, and risk. M&S is just one of many ways that these senior leaders make informed decisions, but the M&S must aid in the communication, support, and understanding of those decisions.

2.3.2 Challenges

2.3.2.1 Complexity

As the battlefield and the interconnected systems-of-systems utilized by the military become increasing complex so too do the associated DoDAF viewpoints. These DoDAF products are an important means to track, compare, and assess systems and changes to systems. These products/models are necessary to document, in detail, the essential elements and characteristics of a system and systems of system. Most importantly, architecture is used by the DoD "to understand what technology is needed and when to invest in that technology" [28]. However, the increasing complexity of large systems can lead to DoDAF products that are so large and so interlinked over so many viewpoints that the resulting products are difficult, if not impossible, for the average decision maker to comprehend let alone visualize the impact of technology or where to implement it.

2.3.2.2 Bookkeeping

Furthermore, with constantly evolving requirements and operations, existing DoDAF products may be incomplete or out of date. In addition, multiple systems that may have to interact on the battlefield are treated separately in terms of DoDAF creation, as they are managed by separate entities. The development of these products can be tedious, and at times are done after-the-fact as a bookkeeping measure rather than as part of systems engineering design. Often this is the result of a misconception that DoDAF architecture products serve as a required bureaucratic exercise with no real long-term use [28]. Clearly this lack of updated and complete documentation presents difficulty when trying to make informed decisions on monetary investments, technology improvements, employment methodology, and the evaluation of alternatives against desired operational MOP and MOE across systems.

2.3.2.3 Static Architecture in a Dynamic World

In large part, this difficulty is compounded by the fact that these products are static whereas real combat systems-of-systems also have high levels of dynamic complexity and combinatorial complexity that arises from multiple systems interactions over time [51]. Decision-making and assessing dynamic systems are cognitive skills. Dynamic decision making requires the ability to process interrelated decisions and subjective uncertainty with the aim of attaining and maintaining a goal state with the ability to explore alternate courses of action and explore the system [52]. The human mind cannot easily synthesize multiple complex, interlinked static models. Static DoDAF viewpoints and models used for the design and analysis of the complex DoD system-of-systems, lack an efficient and holistic executable framework necessary for key decision-makers to rapidly foster exploration and conduct informed, traceable, and repeatable means and ways trades across the DOTmLPF-P spectrum for strategic-level financial, structural, and human investment decisions evaluated against operational measures of performance.

The ability to view executable dynamic simulations and results in real-time to observe how and why things change over time can provide much better insight into what variables influence desired output behavior. This ability is improved further if decision makers can interact and make changes in a hands-on manner [53]. This is one of the reasons the military uses wargames and large-scale simulations to aid decision making. Static architecture cannot provide feedback information that is critical to the learning and decision-making process- especially for complex systems with potential interactions that are beyond human ability to visualize.

DoDAF products are intended to be the foundations systems-of-systems and interoperability. Unfortunately the DoDAF architecture framework does not require inclusion nor provide guidance on a means to create executable dynamic simulation in conjunction with the creation of system architecture [28]. A static MBSE environment doesn’t include trade-off analysis, Monte Carlo simulations, game theory or other complex modern modeling and simulation tools to support analysis [54].

In fact, Dr. Saurabh Mittal [55], of the Arizona Center of Integrative Modeling and Simulation (ACIMS) at University of Arizona identifies six primary shortcomings of static DoDAF, many of which have been discussed thus far. They are as follows:

1) DoDAF mentions “executable architectures” but there is no methodology recommended to facilitate the development of executable DoDAF models.
2) DoDAF overlooks the model-driven development approach. Consequently, there is no formal M&S theory that DoDAF mandates.
3) DoDAF fails to address performance issues at the OV level.

4) DoDAF fails to include measures of effectiveness (MoEs) that can be evaluated at the OV stage. If any performance measures are considered at all, they are at the SV level. System parameters and performance is at a totally different resolution than MoEs.
5) There is no mechanism to perform verification and validation (V&V) at the OV stage.
6) It fails to address M&S as a potent evaluation and acquisition tool.

2.3.2.4 Exclusion of M&S

Executable simulations also take time to create, are often expensive (in terms of computing power and money) and can face questions of validity. The DoD *Capabilities Based Assessment Users Guide*, recognizes this fact and states that the "chain of command knows that analysis, modeling, and simulation can be very time consuming and expensive, and that many huge DoD analyses have produced little or no return" [37]. Additionally, the *DoD Modeling and Simulation Guidance for the Acquisition Workforce*, further provides that the "credibility, i.e., trustworthiness, of M&S is a paramount issue. If M&S cannot provide credible insights, the program is ill-served and the M&S investment wasted" [26]. As a result, simulation can often be excluded in favor of traditional static MBSE products and standardized decision making processes based on assertions and historical precedent, particularly with so many systems being developed under tight time lines by different organizations focused on different programs that, while being developed independently will eventually interact on the battlefield.

2.3.2.5 Poorly Crafted and Supported Requirements

Exclusion of M&S is especially common during the end-user driven capabilities-based assessment (CBA) and development of the initial capabilities document (ICD). (Figure 10) This exclusion is likely contributed to the Army Operations Researchers and Acquisitions observation that the ‘process does not strictly enforce the need for evidence-based assessments” and that “some capability gap rankings [are] based on SME input only that may or may not be informed by a large body of evidence” [13].

A reasonable, supportable estimate and justification of required changes should be made prior to the commitment to a materiel solution and the formal analysis of alternatives. At that point, many more systems, functions, detailed interactions, stakeholders, schedules, and budgets, must be considered as the capability is being developed. The systems and the SoS must be assessed in a representative joint operational environment. Without sufficiently reasonable estimated benefits of operational performance, the only means of assessment is live testing. However, live testing is expensive and “the scarcity of real equipment, range limitations, security, and safety concerns place significant limitations” on the process [26]. Therefore, it is far better to have reasonable assurances on the benefits prior to committing to this phase. The key to successful acquisitions is getting the first step correct: defining requirements and generating the ICD [22]. Getting this correct from complex static architecture is a significant challenge. Poorly written requirements (operational and technical) can lead to failure.

The sponsor/end-user must be able to identify capability gaps in operational scenarios and estimate the impact of those gaps in terms of operational MOP and MOE. Additionally, they should be able to characterize the reasons for the gaps such as policy limitations, insufficient forces, or ineffectiveness of capability in a given scenario.[37] An analytical approach is critical to articulate these gaps and the proposed DOTmLPF-P and requested material solutions. The user must be able to justify which gaps are the most critical and back it up with evidence and be able to articulate it to senior stakeholders to convince them that the CBA built off of DoDAF provides “reasonable estimates for warfighting causes and effects.”[37] With so many potential focus areas and combinations that could lead to desired operational benefit, the end user must have a way to convince and support their request.

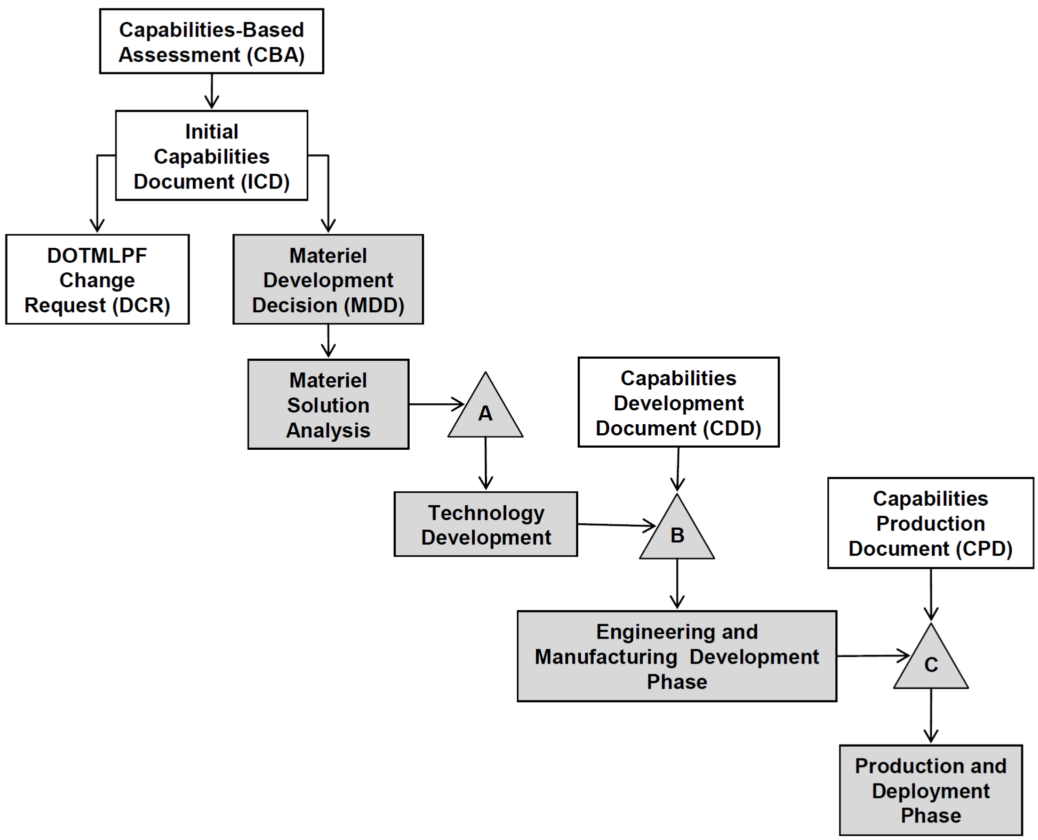


Figure 10: DoD Needs Identification and Solution Process recreated from [37]

The inability to identify key areas upon which to focus technological developments, monetary investments, and material, training, and human resources can have a few potential outcomes. First is the inability to convince key decisionmakers that money should be invested, leading to a failure to invest funding or research into a potentially beneficial technology or DOTmLPF-P solution. Alternatively, money and effort could be invested in an area, based upon solely upon subjective SME assessments and decisions to ensure specified requirements are met. This can lead to event-oriented decision-making that address only immediate concerns without the ability to observe actual benefits, trends, feedback, and effects over time and does not provide the operational benefit for which they had anticipated.

2.3.2.6 Overlooking Elements of DOTmLPF-P

Lastly, if analysis focuses only on the technical or material solution, as is done in the material solution phase (analysis of alternatives) it could fail to capture additional manning, resourcing, and training requirements that are required to augment the material solution to make it fully viable as well as the means by which the resources will be employed. Though the 'm' is intentionally small in DOTmLPF-P to indicate it is only part of the solution and the last resort, history shows that it can often be the first solution at the expense of the other DOTmLPF-P variables.

2.3.3 Gap Identification

Creating executable system architecture is an ongoing area of research and development efforts. The approaches to generate executable architecture have been broad and attempt to capture the latest trends in simulation and optimization techniques. In 2008, researchers from Missouri University of Science and Technology, introduced a paradigm for creating executable architecture for utilizing SysML™ software to translate SysML™ inputs to colored petri nets for dynamic system analysis [56]. In 2011, Dr. Griendling and Dr. Mavris introduced an architecture-based approach to identifying system-of-system alternatives utilizing DoDAF products deemed "ARCHITECT" (Architecture-based Technology Evaluation and Capability Tradeoff) for use in early development of systems of system acquisition efforts [57, 58]. In 2012, researchers at the U.S Air Force Institute of Technology conducted a study of executable architecture from DoDAF utilizing MATLAB's SIMULINK™ to evaluate completeness and effectiveness [34]. Finally, in 2015, researchers from Purdue University proposed a methodology for creating executable agent-based simulations for early phases of systems engineering analysis of alternatives utilizing MATLAB™ to simulate systems-of-systems architecture by modeling each function of the system as an agent [59].

In 2014, Dr. Steven H. Dam, who participated in the development of C4ISR Architecture Framework and the subsequent DoDAF, published his book *DoD Architecture Framework: A Guide to Applying System Engineering to Develop Integrated, Executable Architectures.* In this book he highlights the challenges associated with creating executable architecture, available techniques to create them, and recommended criteria for selecting a technique. In his book, he highlights four major reasons that architecture is often not evaluated as executable: 1) a majority of system architects are not versed in M&S; 2) M&S people don't speak "architecture"; 3) the MBSE community caters to system architecting professionals; and 4) current architecting tools to not allow for robust M&S (namely discrete event simulation, the in-vogue simulation method at that time) [28].

From the previous section, one tends to ponder what aspects of executable architecture could possibly be improved upon. Because of the large disparity of methods by which DoDAF architecture products are created and maintained and for the diverse military functions for which they are created, no single solution is adequate, and none of the touched upon methods suggest they are the ultimate solution. However, a few shortfalls are evidence by real world experience.

The methods are mostly intended to analyze alternatives in early phases of acquisition. The point of analysis of alternatives is too far invested into the commitment of JCIDS process and far too detailed, takes too long to create then simulate, do not account for incomplete documentation nor crossed referenced documentation from other systems. Additionally, the methods allow for technology trades and some attempt to account for tactics and actions of agents, they do not include strategic-level leader considerations across the DOTmLPF-P spectrum such as the employment policy of assets, personnel manning, skills, training, and other difficult to quantify elements of real operations.

As Dr. Dam notes in his book, a system engineer has two primary functions: *technical orientation* (transform requirements into solutions, generate information for decision-makers, and inform follow-on phases) and *management orientation* (informs controls needed to balance all system elements in the architecture) [28]. The methods are data driven and fail to acknowledge the management orientation. Additionally, these methods face the similar challenged that other M&S have faced in terms of their ability to transparently communicate information to stakeholders and decision makers. As noted in the Rand publication, *Making the Soldier Decisive On Future Battlefield*, "[c]urrent systems that claim to take a data-driven approach tend to only recreate a playback of a scenario as opposed to creating an interactive simulation" [60] Many of the existing models, focus on output values as opposed to trends or policy, and tend to be abstract, making them difficult to explain to senior audience which could lead to lack of trust in validity of model outputs, however valid.

Table 3: Desirable Attributes of an Executable Architecture

Desirable Attributes of an Executable Architecture
Easy to understand and communicate, visual and interactive
Able to capture aspects across the DOTmLPF-P spectrum in a comparative, traceable, reproducible manner.
Able to evaluate effectiveness of solutions in an operational environment
Able to be rapidly modified and allow the ability to make trades and play games
Provide insight into second and third order effects due to complex feedback structure.
Computationally inexpensive and rapidly developed so it is more likely to be included.
Produced using existing DoDAF products to develop interactive simulation regardless of format used to create architecture.
Allow for the integration of separate systems with different DoDAF products

Ideally, an executable architecture model that could be constructed quickly and in a manner that is easy to understand and could be created in the early phase of CBA by existing architecture regardless of format and create something that provides a common operating picture, the equivalent of the ubiquitous OV-1 (Operational Viewpoint 1) diagram-something everyone can understand look at and understand. However, unlike that simple picture, the model has underlying mathematical complexity that allows simulation nearly instantaneously with a means to explore millions of trades and can be used to play games to inform policy and acquisition decisions.

2.3.4 Overall Problem Statement

Chapter 1 and the preceding sections of this chapter reviewed the basic principles of the DoDAF and the need for an executable architecture for existing complex systems-of-systems along with the challenges and desirable attributes associated with it. A crosswalk of relevant stakeholders, perceived gaps, and identified need is shown in **Error! Reference source not found.**. From these summaries, an overall problem statement can be developed as follows:

> Static DoDAF viewpoints and models used for the design and analysis of the complex systems-of-systems, lack an efficient and holistic executable framework to foster rapid exploration and conduct means and ways trades across the DOTmLPF-P spectrum evaluated against operational MOP/MOE.

Figure 11: Overall Problem Statement

From this overall problem statement, an overall hypothesis must be proposed. The subsequent sub problems and an case study using Aerial Intelligence, Surveillance, and Reconnaissance (AISR) with reach-back Processing, Exploitation, and Dissemination (PED) with Long-Range Precision Fires (LRPF) system-of-systems in a notional Decide-Detect-Deliver-Assess (D3A) targeting operation will be used to defend the hypothesis to this overall problem statement.

2.3.5 Overall Hypothesis

If a complex system-of-systems is modeled with a holistic executable architecture (EA) it can be used to sufficiently identify the critical elements necessary for further analysis to better assess its overall behavior and structure and inform higher-level decisions.

Figure 12: Overall Hypothesis

It is the authors belief that there is a middle ground fit-for-purpose solution that can be rapidly developed to inform decisions. From the authors perspective, through an objective analysis of commonly available simulation methodologies, System Dynamics provides a "goldilocks" simulation method that can serve as the simulation equivalent of the ubiquitous OV-1 High Level Operational Concept Graphic: something that communicates well to all audiences and provides high-level technical orientation and management orientation.

The objective of this research is to explore an efficient alternative means to developing executable architecture for early capabilities needs assessment on existing systems-of-systems There is a need for a new approach to enable to efficiently assess critical systems that serve as the 'brain center' of the military and encompasses hundreds, if not thousands, of interacting systems yet are too large to envision without executable architecture yet so large that traditional methods are too time consuming and computationally expensive.

2.3.6 Overall Research Objective

Develop and demonstrate an efficient and holistic framework for a complex system-of-systems to allow for means and ways trades and enable multiple stakeholders to conduct electronic design reviews on an existing system-of-systems in order to analyze technological benefits, limitations and policy impacts for future investment and strategy.

Figure 13: Overall Research Objective

This method will demonstrate the ability to aid the modelers and decisionmakers in the identification and implementation of feedback systems thinking and reduce complexity of an existing complex system-of-systems architecture that is constantly evolving. This method can be used to visualize dynamically complex problems or scenarios wrought with interdependencies and simplify them with the aim of qualitatively and quantitatively evaluating the effect of decisions and policy on measures of performance or effectiveness for a given operational task while testing new policies and strategies. This method allows for human decision-making but also provide an insight into emergent behaviors of new techniques and tactics to refine assumed relationships for better understanding and focus.

2.4 Problem 1: Evaluating Existing Complex System of System Architecture

During primary development, the C4AISR Architecture Framework and the later DoDAF models considered only computers and software. Therefore, M&S were of discrete events and thusly the discrete modeling paradigms (Petri-Nets, DES, etc.) were the default for executable dynamic models [30, 56]. However, this notion fails to account for the emergent behavior inherent to the assets that collect the intelligence in real-world operations and assumes linear relationships between variables with a fixed structure. Real-world military systems-of-systems are more complex. They include not just in the employment of the assets but in the manpower requirements to operate equipment and to conduct operations. Additionally, traditional methods neglect the possibility of enemy actions to disable the network or eliminate assets. Rather than evaluate a system-of-systems' effectiveness against stated performance requirements for a system design specification, it is useful to mapped to operational MOP & MOE to better evaluate the ability to satisfy the warfighters' needs.

Current proposed means to develop executable architecture are cumbersome, manually created simulations to assess specific elements of architecture connectivity and design MOP.

Figure 14: Problem 1 Summary

Existing DODAF models are static and while they can be updated over time to reflect changes, most changes that are made to an existing system-of-systems are difficult to assess. Recommendations for changes usually come from gaps identified by end users through operational use though no means exists to efficiently evaluate changes of policy and technology improvements to system structure to the impact to larger DOTmLPF-P considerations against operational force constraints and against operational MOE/MOP.

While methodologies such as the Architecture-base Technology Evaluation and Capability Tradeoff (ARCHITECT) have been proposed to improve agility in defense acquisitions these methods, like the DODAF models themselves are aimed at the early phases of design and acquisition. Dr. Kelly Griendling, creator of ARCHITECT, stated that "a better treatment of both the consideration of the ease of integration of solutions into the existing SoS and exploration of verification are both also areas which would benefit greatly from further research"[58].

Demonstrate the ability to efficiently execute and evaluate an existing dynamic system-of-systems architecture (via AISR PED use case).

Figure 15: Objective 1

2.4.1 Research Question 1.1

What means are available to aid in the use of systems thinking to understand the system structure as the first step of conceptualizing the model?

Figure 16: Research Question 1.1

Typically, the construction of a model and simulation can be accomplished through a combination of data, observations of an existing system and developing the system structure through interviewing SMEs. Military systems have the unique requirement of MBSE DoDAF models that can be used as the foundation of the creation of the model, even if not all DoDAF viewpoints may be available or even used for an existing system.

2.4.2 Hypothesis 1.1

Existing DoDAF models of a system or system-of-systems can be used to initially describe and construct the fit for purpose executable architecture. Interviews with SMEs and data can then be used to close the gap and adjust to real-world dynamics.

Figure 17: Hypothesis 1.1

However, as discussed in Sections 2.1.3 and 4.3 on DoDAF architectures, there are over 50 architecture viewpoints/models, only some of which exist for the current system.

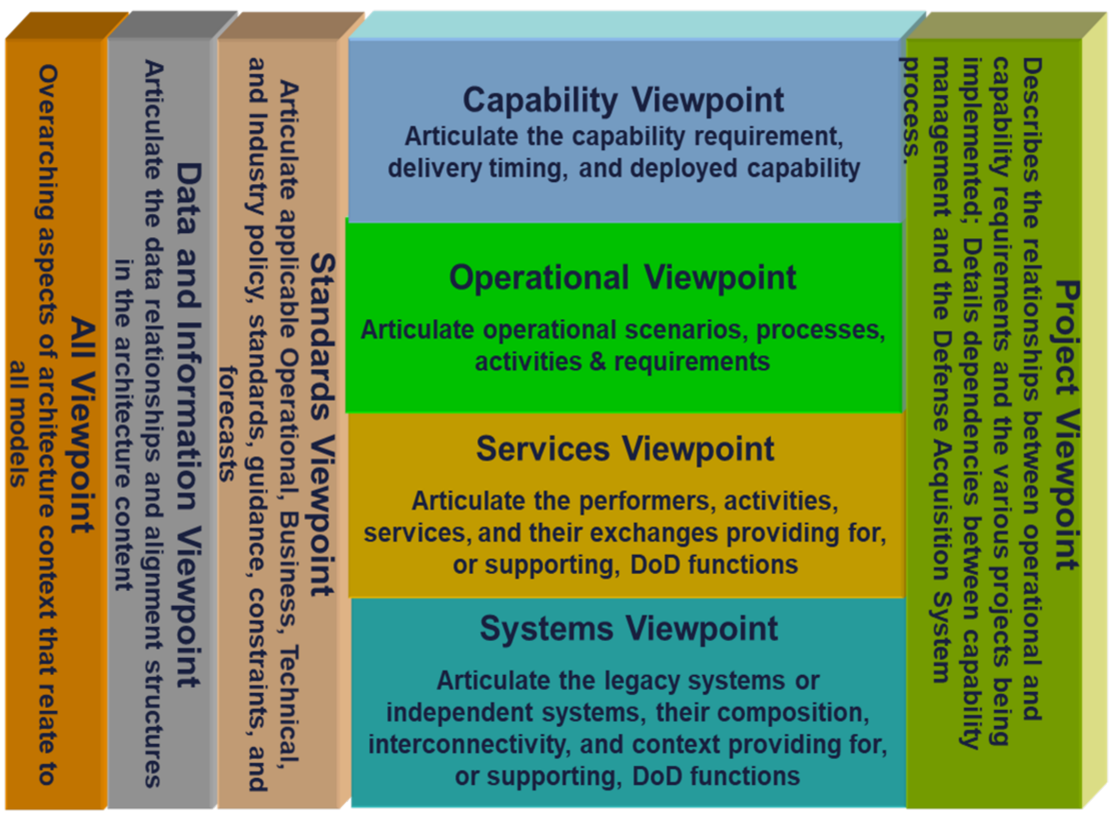


Figure 18: DODAF 2.0 Overview from [31]

This fact raises the question of which of the 50 DODAF models are needed and can be mapped to a M&S. Section 4.3 will discuss which viewpoints are most valuable for the creation of the executable architecture and the rationale for use.

2.4.3 Research Question 1.2

Which simulation methods are suitable to create an executable architecture of a complex system-of-systems?

Figure 19: Research Question 1.2

Section 3.3 will provide a review reviewed the most common modeling and simulation paradigms. Section 2.2 will provide a review of military modeling and simulations history, the philosophy behind military, simulation types currently in use, and most importantly, why the military uses the methods it does and the benefits of each. The answer may not be a single method; it may be combination of methods. It certainly depends on the questions being explored.

Additionally, the DoD *Capabilities-Based Assessment Users Guide* stresses that the approach must fit the problem rather than be driven by availability of a particular tool or any claim that an existing model is already validated [37]. Dr. Seth Bonder, emphasized that statistical models that use data are intrinsically tied to current systems and are best used in retrospective analysis and that process models of relevant phenomena will better facilitate prospective analysis (i.e., planning and prediction). Models of military operational phenomena are tools with which to "provide useful information and insights to assist decision makers in areas where direct experimentation is expensive and, many times, impossible" though they are "not accurate predictors of the future" [38].

The model should ideally be able to capture aspects such as human behavior, reliability, and emergent behaviors of complex systems. It should be interactive to inform decisions and provide insight rather than provide a simple playback. It should allow rapid trades of means and ways. It should be traceable, repeatable, and rigorous. It should be reduced to the minimum complexity required to evaluate the specific problem for which it is being applied. It should be inexpensive both in cost, in computing power required, and in time as the political environments, threats and funding can change rapidly. It should be easily modified to evaluate different structures and policies. The proposed model should be able to identify critical elements of the system that contribute most to overall MOE and MOP to identify current shortfalls and inform future acquisitions and DOTmLPF-P solutions or recommendations.

2.4.4 Hypothesis 1.2

System Dynamics can provide an overarching M&S architecture that can capture key aspects of the system and enable understanding of technology benefits and limitations, policy impacts, and the likely outcome of future investment strategies.

Figure 20: Hypothesis 1.2

To construct this M&S, we must first seek to apply systems thinking. Regardless of the modeling methodology pursued, we must first seek to understand the system structure so that we can recognize interconnection and identify and understand feedback which is also part of the first step of Forrester's modeling process (Figure 47: Forrester's 6-Step SD Modeling Process recreated from [69])

2.5 Problem 2: Changes to Policy and System Structure

> Decision-makers need a means to identify are the most influential elements in obtaining the desired performance over time relative to operational MOPs/MOEs to properly allocate resources and assets with structural changes.

Figure 21: Problem 2 Summary

Stakeholders are under constant pressure to prioritize technological improvements and investments. It is difficult to ascertain, especially in complex systems-of-systems, if existing policies, structures, assets and resource allocations (training, personnel, monetary investments, etc.) will have a large, if any, desired benefit to actual operations This can be further complicated by changes to actual structure of a system-of-systems which is more difficult to capture in many M&S paradigms without major revisions.

> Demonstrate the ability of the executable architecture to assess variations in system structure and holistically integrate various DOTmLPF-P solutions for sensitivity comparison.

Figure 22: Objective 2

2.5.1 Research Question 2.1

With limited data available to assess system-of-systems architecture, what method provides the best means to determine which elements of the system are most influential?

Figure 23: Research Question 2.1

For many DoD SoS (particularly in AISR and PED), obtaining accurate statistical data can be difficult to obtain, can be unreliable as it is predicated from the presumption of accurate unit recording and reporting, or may even be classified. Therefore, it can be difficult to ascertain which elements of the system-of-systems is the most influential so that structural changes and effects can be evaluated. The methodology must allow for variability of input values.

2.5.2 Hypothesis 2.1

A Monte Carlo Simulation for univariate and multivariate sensitivity analysis (parametric analysis) can sufficiently determine the sensitivity of the model behavior to variations in parameter values.

Figure 24: Hypothesis 2.1

Previously the idea that sensitivity analysis via Monte Carlo Simulation has been identified as lacking in static architecture MBSE. For the interests of this study, the intent is to demonstrate the framework as an ability to analyze existing systems-of -systems with a use case of AISR PED at the open source UNCLASSIFIED level. Therefore, actual data will not be used, but estimated values that can then be varied parametrically to demonstrate dependencies and influence of input variables on the system behavior and values over time.

Monte Carlo analysis is a technique that allows the model builder to randomly assign input values based upon a probability distribution assigned to said input variable to account for epistemic uncertainty. By running tens of thousands of cases, it allows the simulation to output the results a histogram and can be displayed as probability density functions (PDFs) and/or cumulative distribution functions (CDFs). This also allows the modeler to take a deterministic model and make it into a probabilistic model that can probabilistically display the results over time.

2.5.3 Research Question 2.2

Changes in structure and policies can add difficulty in the creation of an executable architecture. For the use case, the introduction of the MDTF creates an alternative structure. Decision makers need to determine how information and resources should be tasked to the MDTF PED system to improve effectiveness relative to operational MOPs/MOEs.

Figure 25: Problem 2.2 Summary

An executable architecture that requires a complete reconstruction of the simulation for changes in structure and policy (which are common) is less likely to be utilized due to cost and time constraints. Flexible modifications of the executable architecture are paramount for rapid strategic-level decision making. Section 4.2.2 will introduce Army's development of the Multi-Domain Tas Force (MDTF) and the intent to integrate the

MDTF into the existing AISR PED architecture as a means to expedite actionable intelligence from the sensor to the shooter to effectively engage enemy targets. While the Defense Information Systems Network (DISN) already provides a way for tactical and operational intelligence analysists to access UAS feeds and posted intelligence products from the federated PED (see Appendix A) the experimental unit, in an attempt to combat fast moving multi-domain operations, intends to have its own PED internal to the organization to rapidly process information on actionable targets to form quicker targeting packages while the primary PED process larger-scale, specialize PED for long-term target development and strategic intelligence fusion. However, this begs questions such as: '*How should personnel be allocated to this new structure? How much intelligence must be routed through the MDTF PED?; 'Will such a change in structure increase or decrease the amount of intelligence products processed?';* and *"How much growth (DOTmLPF-P) is required*?"

Using the AISR PED enterprise as a use case, demonstrate the ability to develop an alternative system structure to include the MDTF and determine how much intelligence it must be able to process to increase the operational effectiveness.

Figure 26: Objective 2.2

For the AISR PED use case, how do the additional structure of the MDTF and additional DOTmLPF-P changes affect the PED system? How should resources be allocated?

Figure 27: Research Question 2.2

This research question will serve to demonstrate the ability to vary system structure in the executable architecture to evaluate its effect on the desired performance over time against operational MOP/MOE. Trivial solutions would include simply increasing the number of personnel and assets, but the real-world is constrained.

2.5.4 Hypothesis 2.2

Adding the MDTF will improve the operational MOP/MOE by reducing delays only if it is allocated the appropriate number of personnel and technical improvements.

Figure 28: Hypothesis 2.2

The major contributor to dynamic behavior is a result of delays.[51, 61] Changes to SoS structure and policy can have negative effects despite the best intentions of the decision makers. There is a need to inform policy and structural decisions in an executable fashion. Adding complexity to the structure could serve to create additional bottlenecks of reporting if the appropriate information is not routed properly to the expedited system. Additionally, DOTmLPF-P considerations will have a direct impact. For example, if manpower and workstations are simply transferred from the federated PED to the MDTF, the result may be delayed strategic products and increased intelligence backlog at the federated PED. An executable architecture allows decision makers to explore alternatives in an interactive manner to visualize the effects.

2.6 Problem 3: Asset and Resource Allocation

2.6.1 Research Question 3

Event-oriented thinking that is commonplace within large organizations and is very typical in decisive minded leaders. This is because the real-world environments in which they must make decision is one of bounded rationality where information is limited and often unreliable, the human mind is limited in capacity to evaluate and process the available information, and there is a limited amount of time to decide. This just as true in combat operations as it is in the ever changing geopolitical and fiduciary landscapes. dissemination, tactical relays, satellite gateways, network operations, and lastly, platforms.

Can the executable architecture be used to identify elements and values of the SoS architecture that have the greatest impact on operational MOP/MOE in the larger operational construct??

Figure 29: Research Question 3

For the use case, the AISR PED enterprise architecture use case will continue to be under significant strain to satisfy increasing demands for intelligence used for planning and targeting. The Unmanned Systems Integrated Roadmap 2017-2042 identifies four primary areas of emphasis for UAS improvement: interoperability, autonomy, network security, and human-machine collaboration [11]. The DOD has proposed improvements to the enterprise in five blocks through FY2024 depicted in the Table 4 and Figure 30.

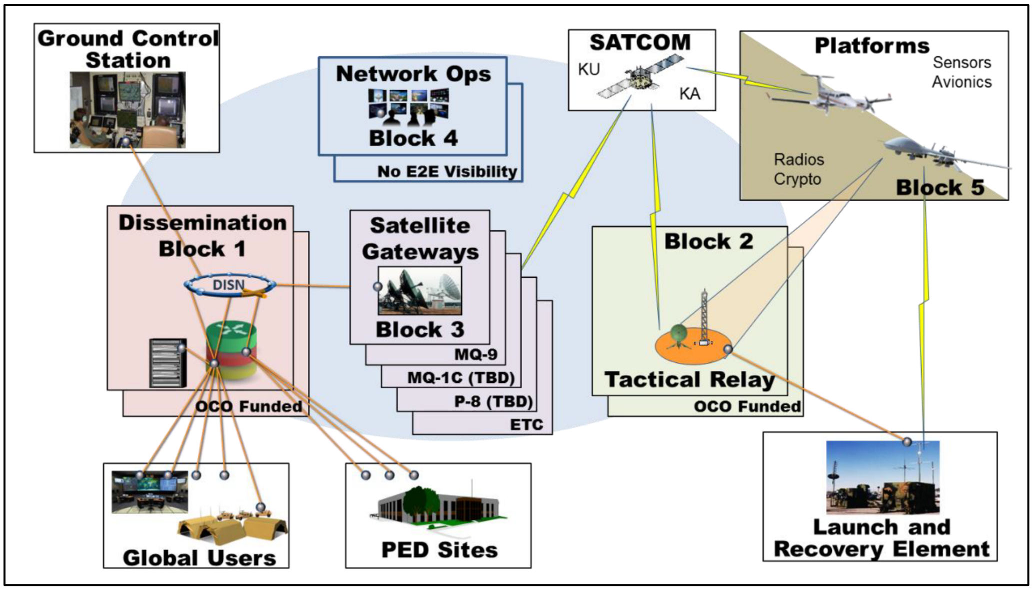


Figure 30: UAS Integration Roadmap Schematic from [11]

The UAS Roadmap phased materiel initiatives provide an ideal example of the urgent need to rapidly evaluate proposed materiel solutions along with other elements from across the DOTmLPF-P spectrum for complex systems of systems. Stakeholders must be able to prioritize requirements and asset allocations to meet the increasing demand in a fiscally constrained environment.

The reader should refer to Appendix A that outlines the addition of UAS assets occurring throughout this entire period This linear treatment implies that some thought process was put in to prioritizing these efforts; it could be limited by budgetary constraints or technology availability. However, as this information is not readily available to the public and DOD *Unmanned Systems Integration Roadmap 2017-2042* provides little detail as the reasoning behind the sequence of the phased block improvements. Therefore, one could also presume, based on experience in the field, that the order is driven by demands from end users who are identifying sources of their particular “bottle necks’ without looking at the holistic system.

Table 4: UAS Integration Roadmap Initiatives from [11]

Block	**Element**	**FY**	**Purpose/Plan**
1	Dissemination	21	Programmed capability to ingest and distribute sensor/video data in near real-time on US classified networks
2	Tactical Relay	22	Add LOS-only platforms to Defense Information System Network (DISN)for global distribution
3	Satellite Gateways	23	System of universal gateways will support global distribution of sensor data via DISN
4	Network Operations	24	Integrate current disparate NETOPs activities and establish COP for sensor data transport
5	Platforms	24	Life-cycle upgrades, multi-band satellite capabilities, improved survivability in A2AD, NSA encryption to prevent intercept/tampering

Obviously, the block improvements consider the entire system, but over a fiscal year timeline. This begs the question *‘what if money runs out or the budget is cut?’* as often occurs in the PPBE cycle. There is also a question of if the U.S. will get involved in another global engagement or escalate demands in one of the other areas in which it is already operating. *Which of these initiatives are the most influential on the behavior of the system-of-systems and should be prioritized if technology allows?* These block initiatives exist because decision makers and end users need more AISR than is currently available. The *Roadmap2017-2042* refers to machine learning and artificial intelligence throughout the document as the largest initiative to improve PED. *Will PED technology alone improve the* *system-of-systems and how much (in terms of amount of intelligence processed resulting reducing delays from sensor to shooter) is needed?*

2.6.2 Hypothesis 3

The EA can be used to generate surrogate models to identify the most important aspects and optimal values of the SoS variables against operational MOP/MOE to inform exploration and investment.

Figure 31: Hypothesis 3

2.7 Research Intent and Summary

The intent of this research is to explore an efficient alternative means to developing executable architecture for early capabilities needs assessment on existing systems-of-systems. Given the authors extensive background in aerial intelligence, surveillance, and reconnaissance, and the C4ISR origins of the DoD Architecture Framework, such a use case to demonstrate this methodology seems appropriate.

This method must be able to holistically evaluate higher-level DOTmLPF-P considerations for the current, transition, and future states of the enterprise. It should provide decision makers with insight as to the trend effects resultant of decisions, operating methods, and enterprise structure. It should be capable of being used as a training tool much like war gaming; a laboratory to play games and foster explorations. It should aid in identification of vulnerabilities and potential for improvement. Ideally the simulation would be computationally inexpensive. The results should be traceable and repeatable to provide scientific backing to requirements and decisions.

An executable dynamic simulation is often needed to fully assess, visualize, and understand a complex architecture framework, yet DoDAF guidance provides little to no guidance on executable simulation. The primary simulation methods used in modern military simulations tend to be computationally expensive which leads to lack of use or limits use to subsystems analysis. As a result, simulation developers may fail to capture the complexity of the system-of-systems or may not fully explore the design space to evaluate many possible alternatives across the DOTmLPF-P spectrum against operational MOP/MOE. Other methods include the use of war games, expert seminars/workshops, or the infusion of experimental organizations into costly large-scale exercises. These methods provide valuable insight, but the results are not repeatable. A “middle ground” between no simulation and highly detailed simulation must exist that can adequately enable an executable architecture capable of providing technical orientation and management orientation [28].

The power of modeling is its ability to aid in the identification of macro-interaction and trends expected of systems-of-systems and/or be used to inform high-fidelity simulations. The intent of this study is to demonstrate the applicability of modern computer assisted methods to assist in providing previously difficult or impossible to identify dynamic and complex second and third order feedback loops at the macro-level (lower fidelity). In addition, this method will aid in identifying areas where it may be necessary to utilize higher fidelity engagement-level modeling as necessary to inform only the aspects of the aggregate model that are most influential on the desired measures of effectiveness.

By doing this, modelers can generate a model using available expertise and information rather than rely solely upon historical data that was poorly collected, does not exist for future systems, or may be unavailable due to classification levels.

The author's intent is to develop a system dynamics (SD) model depicting the AISR PED as part of the D3A (decide-detect-deliver-assess) [62] kinetic targeting cycle with long-range precision fires (LRPF) as a use case to demonstrate the overall method. The author desires to yield new observations, provide traceability and reproducibility while examining the consequences of proposed structural and technological changes suggested by the Unmanned Systems Integrated Roadmap 2017-2042 to the existing system-of-systems.

The initial model will be based upon the current AISR architecture that has been rapidly expanded over the past 15 years in support of contingency operations. To maintain the model at the UNCLASSIFIED level, a hypothetical multi-intelligence UAS is used with estimated ranges of values. Future models can include the use of classified data. The model will be verified and validated based upon the output trends through direct structure tests (assessing each of the relationships individually) and structure-oriented behavior tests (running the simulation for the entire model). The simulation will then be used to explore sensitivities/primary influencers where small changes lead to large changes to mission effectiveness and substantial return on investment. Monte Carlo analysis will yield multiple combinations that will yield desirable effects which can be used to allocate resources and inform key decision makers. Surrogate models will be created to provide instantaneous results for rapid comparison of various combinations.

Next, the M&Sl, using the existing AISR PED architecture as a baseline, will explore the effects of introducing a structural change to the PED process, i.e. the introduction of the MDTF PED. This model could be used to inform MDTF requirements and structure which can be verified using future exercises (outside the scope of this research). The M&S can be used as the overarching model to play games. For areas determined most influential, smaller higher fidelity models can be used to create a better informed a blended model. (Future work beyond this thesis).

The AISR architecture and sub-architectures are quite expansive and complex. A model that includes every detail would be both classified and take a dedicated team several months to develop. The intent of this academic effort is not to suggest a complete overhaul of the existing reach-back PED system, nor does it intend to demonstrate the complete system with all its complexities. Such an endeavor would be beyond the scope of a single thesis. There is no attempt to make an absolute judgement on any policy, strategy, or acquisition effort. Nor is it an attempt to redefine the entirety of JCIDS and DoD Architecture Format (DoDAF), but rather introduce a way to evaluate trades in a dynamic environment for a large system-of-systems against operational MOP/MOE to improve decision making.

Lastly, the intent of this study is to propose a means to capture aggregate trends associated with difficult to quantify and difficult to model impacts (cyber, training, morale, etc.) that are difficult or impossible to model with high-fidelity simulations reliant upon pre-programmed behaviors. Ideally this simulation is possible with readily available modeling software that can be generated and manipulated more rapidly without the need for massive amounts of computing power required of most modern military agent-based simulations. The author seeks to add an additional methodology to the repository of effective analysis techniques.

An abbreviate summary of the research problems and questions is shown in Figure 32 below for reference.

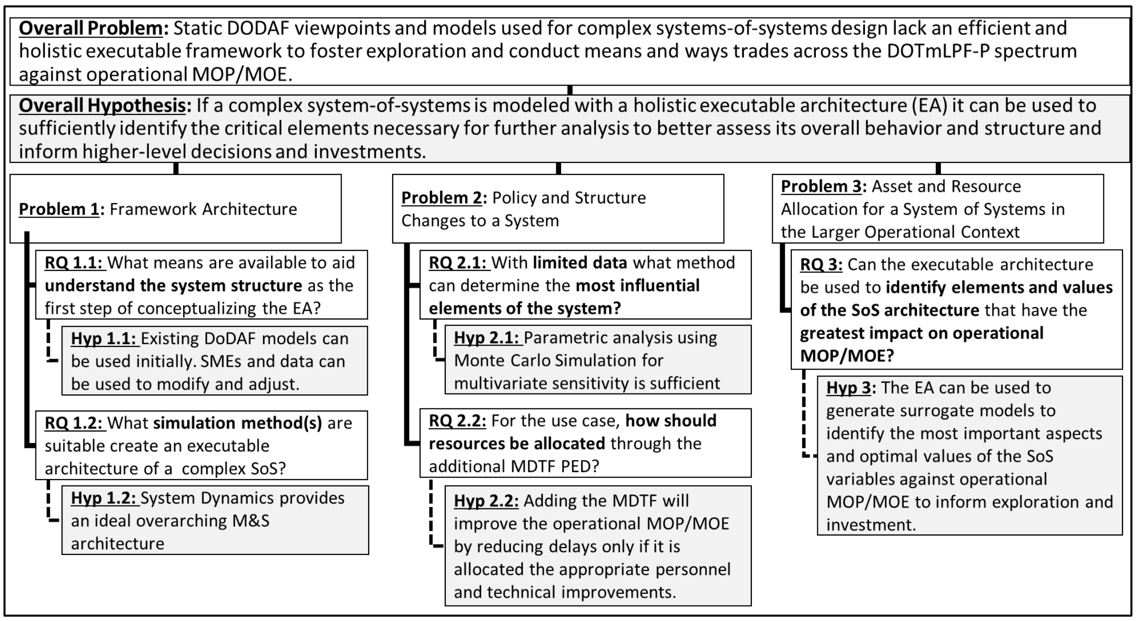


Figure 32: Overall Research Problem Diagram

CHAPTER 3. LITERATURE REVIEW

3.1 Systems Engineering Concepts

This dissertation hopes to explore and demonstrate a solution to the problem of executable architecture through the application of “Systems Thinking,’ Modeling and Simulation (M&S), and surrogate modeling. Before addressing the identified problems and research questions, it is important to provide the reader with some basic understanding of terms and concepts that will be used in this document for reference. This subchapter will address basic Systems Engineering Concepts gleaned from an expansive literature review conducted by the author.

3.1.1 Definitions

3.1.1.1 System

A system is a term that is often used and as a word itself, has been extensively discussed and analyzed. From the *DOD Modeling and Simulation (M&S) Glossary* a system is:

> “A collection of components organized to accomplish a specific function or set of functions” [63].

From the DOD *System Engineering Guide for Systems-of-systems* a system is:

> "A functionally, physically, and/or behaviorally related group of regularly interacting or interdependent elements; that group of elements forming a unified whole” [64].

Lastly, from Dr. C.E. Dickerson and Dr. D.N. Mavris's textbook, *Architecture and Principles of Systems Engineering*:

> "A system is a combination of interacting elements integrated to realize properties, behaviors, and capabilities that achieve one or more stated purpose(s)" [29].

From the above definitions, there are a few commonalities such as interacting components/elements with an intended purpose. With these definitions in mind, a summarized definition is proposed for use in this thesis:

Definition: A **system** is a set of interacting components organized to achieve a stated purpose

3.1.1.2 Terms That Describe a System

It is also important to define terms used to describe a system in terms of modeling and simulation as opposed to those used in physical systems. Physical systems include elements and the interconnections between elements, both of which are self-explanatory. For system models, it is important to define additional terms which are summarized in the Table 5: System Modeling and Simulation Terms .[65]

Table 5: System Modeling and Simulation Terms

System Term	Definition
Environment	Anything outside of the designated system that may affect the system
Boundary	Separates the system from its environment
Entity	Object of interest in a system
Attribute	Property of an entity
Activity	Time period of a specified length
State of a System	Collection of variables necessary to describe the system at a given time
Event	An instantaneous occurrence that may change the state of the system
Endogenous	Activity and events that occur in the system
Exogenous	Activities and events in the environment that can affect the system
Discrete	State variables change at distinct points in time
Continuous	State variables change nonstop over time

3.1.1.3 System-of-systems (SoS)

The definition of a 'system' gives way to the often confused and debated term 'system-of-systems.' The DOD *System Engineering Guide for Systems-of-systems* defines a system-of-systems as:

> "A system-of-systems is a set or arrangement of systems that results when independent and useful systems are integrated into a larger system that delivers unique capabilities " [64].

The 2018 DOD *Manual for the Operation of the Joint Capabilities Integration and Development System* adds the following to the DOD definition:

"SoS may deliver capabilities by combining multiple collaboratives and independent-yet-interacting systems. The mix of systems may include existing, partially developed and yet-to-be designed independent systems" [27].

The preceding definitions represent the commonly accepted understanding of a system-of-systems, namely that the combination of independent systems into a larger system produces a capability that cannot be obtained by the independent systems independently. In 1998, Dr. Mark W. Maier, author of *The Art of Systems Architecting*, proposed a more thorough definition, taxonomy and set of principles to define a 'system-of-systems.'

Taking the commonly accepted requirements of the time, geographical distribution, emergent behavior, and evolutionary development processes, he added two distinguishing principles that are required to apply the term 'system-of-systems': 'operational independence of the components' and 'managerial independence of the components.' Without these two principles Maier posits that a system cannot be a system-of-systems irrespective of emergent behavior, geographical distribution, or evolutionary development processes. Operational independence of the components implies that if the system of system was divided into its individual systems, each would be able to operate independently to fulfill its purpose on its own. Managerial independence of the components means that the can and do operate independently and component systems are separately acquired and integrated [66].

Definition: A **<u>system-of-systems</u>** is a set of interacting components organized to achieve a stated purpose that demonstrates operational and managerial independence of component systems as well as geographical distribution, emergent behavior, and evolutionary development processes.

3.1.2 Systems Thinking

Model-Based Systems Engineering is based on the principles of 'Systems Thinking.' To address the problem of creating executable architecture, it is important to first define what, exactly, "Systems Thinking" is. The term 'Systems Thinking' was coined by Dr. Barry Richmond, a former graduate student of Dr. Jay W. Forrester at the Massachusetts Institute of Technology and a professor of system dynamics at Northeastern University and Dartmouth in 1987. In his words:

> "As interdependency increases, we must learn to learn in a new way. It's not good enough simply to get smarter and smarter about our particular 'piece of the rock.' We must have a common language and framework for sharing our specialized knowledge, expertise and experience with 'local experts' from other parts of the web. We need a systems Esperanto. Only then will we be equipped to act responsibly. In short, interdependency demands Systems Thinking." [67]

As expected, the use of term without a defined architecture, has led to a highly contested definition amongst Systems Engineering experts and no widely agreed upon definition. Figure 33 illustrates the three predominant interpretations of systems thinking.

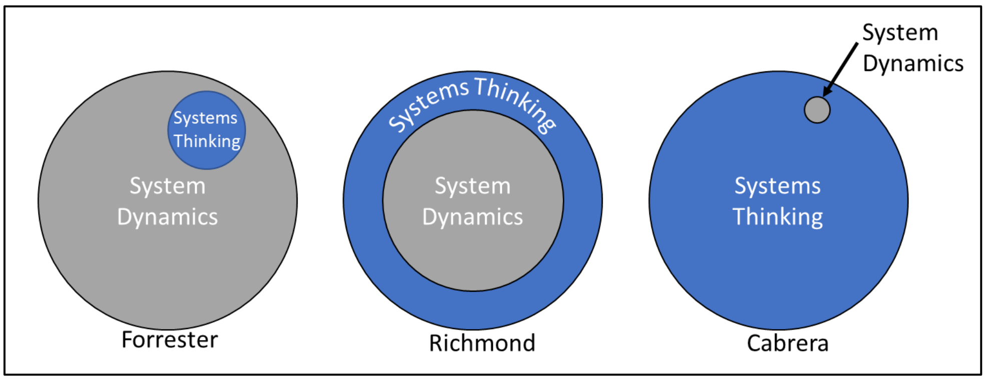


Figure 33: Systems Thinking Interpretations adapted from [68]

The leftmost image is Forrester's definition. From his perspective systems thinking is a minor subset of system dynamics (See Section 3.3.4). Forrester's belief is that systems thinking is little more than a "general and superficial awareness of systems."[69] Though it can be useful in providing the general public of the benefits and importance of systems and provide a "door opener" to system dynamics. As a former Electrical Engineer, he felt that systems thinking and its reliance on casual loops lack the discipline to the thinking that is imposed by level and rate diagrams used in system dynamics. He did, however, recognize causal diagrams as useful after modeling for presenting an "overall impression of [a] subject" to leadership and people who were less interested in the specifics of dynamic behavior as demonstrated by Dr. Peter Senge in his book The Fifth Discipline.[69, 70] In his view, systems thinking is only 5% of the way towards a genuine understanding of system, the other 95% coming from rigorous system dynamics structured models and simulations.[71]

The center image of Figure 33 represents Richmond's view of systems thinking. In this view, systems dynamics is a subset of the larger systems thinking perspective, albeit the central and predominant aspect. Richmond recognized the need for a new way of thinking. While rigorous structuring and simulations are necessary, the models and simulations are not the sole end but rather a means. As a new paradigm, the purpose was to "think more productively about how to improve the way a system worked." [71] The premise being that systems thinking should enable all people to generate insights rather than being limited to skilled systems dynamics practitioners.

Ultimately, Richmond defined "systems thinking" as:

> "[T]he art and science of making reliable inferences about behavior by developing an increasingly deep understanding of underlying structure."

He summarized the pieces of this definition with the diagram shown in Figure 34.

Systems Thinking Is a..	
Paradigm	Vantage Point
	Set of Thinking Skills
Learning Method	Process
	Language
	Technology

Figure 34: Components of Systems Thinking (adapted from [71])

In terms of a paradigm, the vantage point determines how one positions oneself relative to the system to see both the "forest and the trees." While thinking skill are those that one employs to determine what in the system to attend to versus ignore and what to make of it all. In short, systems thinkers see "both generic and specific" and "both the pattern and the event." Richmond postulates that systems thinkers use three thinking skills: system-as-a-cause, closed-loop, and operational thinking. '*System-as-a-cause thinking' is the idea that the structure of a system, rather than outside factors, that causes problems.* Closed-loop thinking is the idea of how the structure is arrayed in reciprocal relationships. Operational thinking, therefore, is using stocks and flows to define that of which the closed loops are composed [71].

The rightmost image in Figure 33 represents a more current paradigm of modern systems, led by Dr. Derek Cabrera at Cornell University. Cabrera's supposition is that most people have adopted the idea that systems dynamics and system thinking are equivalent; analogous to one thinking that hammering nails is the equivalent of being a carpenter. One former is a learned ability to use a tool the latter being an expertise. He also emphasizes that while systems dynamics is a valuable tool, like a carpenter, it is not the only tool in the toolbox. Furthermore, like a tool, while there are many uses for which SD, a powerful framework, is appropriate, there are many uses for which it is not. Knowing when and how to use a tool is the expertise (thinking skills) come into play.[68] Reflected in Figure 33, Cabrera's view is the exact opposite of Forrester, in that System Dynamics is but a minor subset of a much larger body of holistic systems thinking.

The three definitions of 'systems thinking' are not all inclusive, but rather demonstrate the extreme limits of the term and the intent of the originator of the term. Many definitions and key elements of systems thinking have been identified by leaders in the systems engineering field since Richmond first coined the term in 1987, supported by numerous studies. In 2014, researchers, Divya Vuhra Behl and Susan Ferreira from The University of Texas at Arlington, Systems Engineering Research Center compiled a summary of 21 key elements of individual systems thinking identified by 12 of the leaders in the field. They subsequently analyzed the various phraseologies for similarities and developed the relationships between the resulting 21 individual systems thinking elements [72]. All the elements are inextricably linked to the ostensive definition of systems thinking—the ability to think about a system as a whole. A sample mapping of the elemental relationships is shown in Figure 35.

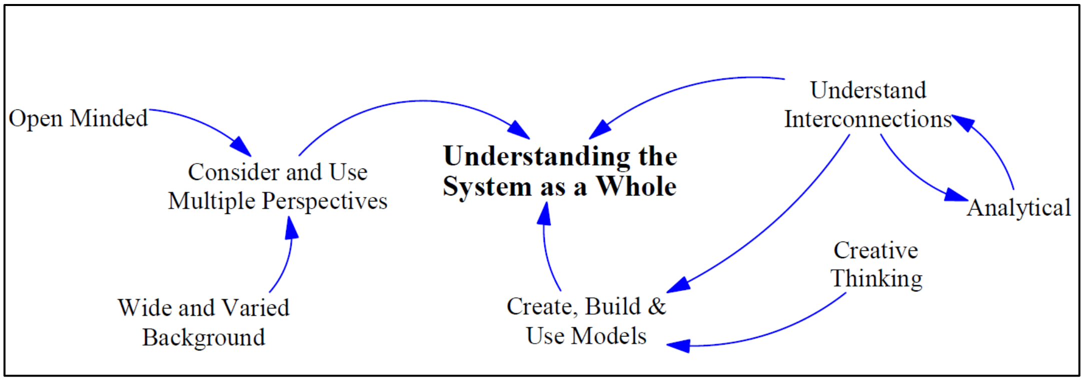


Figure 35: Systems Thinking Element Relationships adapted from [72]

In the level beneath holistic thinking, is are the remaining three elements that most systems scholars agree upon: the ability to understand interconnections; *the ability to create, build and use models; and the ability to consider and use multiple perspectives*. All other elements are linked through these three.

Recognizing the increasing interdependence of complex systems in today's world, and in order bring systems thinking out of the educational margins and into the mainstream lexicon to solve complex problems, researcher Ross D. Arnold and Dr. Jon P. Wade defined systems thinking using a systems approach. Under this approach, they developed a system test to examine eight definition of systems thinking by the 14 most predominant systems thinkers (Richmond, Senge, Sweeney, Sterman, Hopper, Stave, Kopainsky, Alessi, Davidsen, Squires, Wasde, Dominick, Gelosh, and Forrester) as a necessary, but not sufficient criteria of definition completeness. Each definition (like a systems) was broken into its function, purpose, or goal, its elements and the interconnections between them.

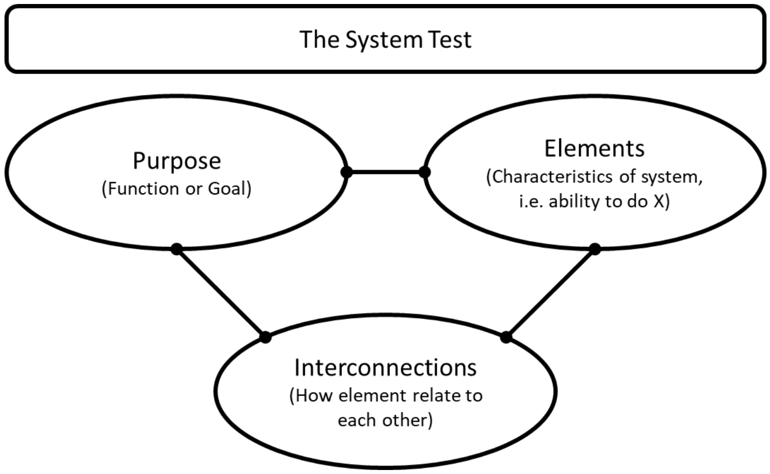


Figure 36: Arnold and Wade Systems Test [67]

Through their examination, they found that every published definition of systems thinking failed the systems test. By their estimate, all definitions focused on the elements of systems thinking but not what systems thinking is or does (the purpose and the interconnections). Arnold and Wade then composed a diagrammatic comparison of the definitions and their various aspects shown in Figure 37.

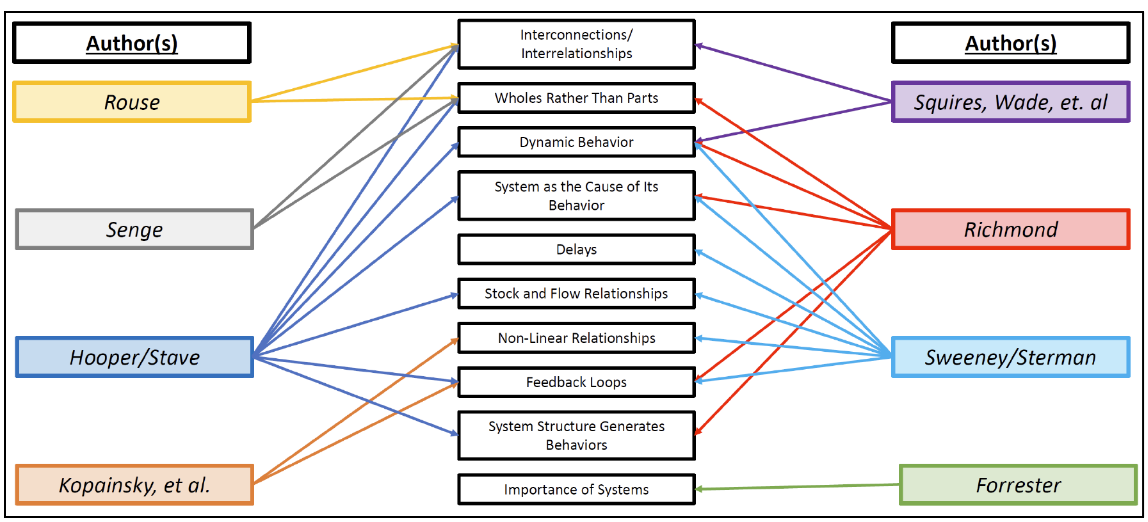


Figure 37: Comparison of Systems Thinking Definitions from [67]

Like Behl and Ferreira, Arnold and Wade found that the most accepted elements of systems thinking were the considering *"wholes rather than parts"* and "*interconnections and interrelationships?*" However, unlike Bhel and Ferreirra, Arnold and Wade found equal concurrence on the elements of; '*dynamic behavior'* and '*feedback loops'* closely followed by the idea *that system structure causes behavior*, much like Richmond. They then crafted a definition of systems thinking that can satisfy the systems test:

> "Systems thinking is a set of synergistic analytic skills used to improve the capability of identifying and understanding systems, predicting their behaviors, and devising modifications to them in order to produce desired effects. These skills work together as a system." [67]

It is this definition of systems thinking that will be used in this thesis.

The set of synergistic skills are as follows:

- Recognizing interconnections
- Identifying and understanding feedback.
- Understanding system structure
- Differentiating types of stocks, flows, and variables
- Identifying and understanding non-linear relationships
- Understanding dynamic behavior
- Reducing complexity by modeling systems conceptually
- Understanding systems at different scales.

Arnold and Wade then interconnected the elements through the following systemigram in Figure 38.

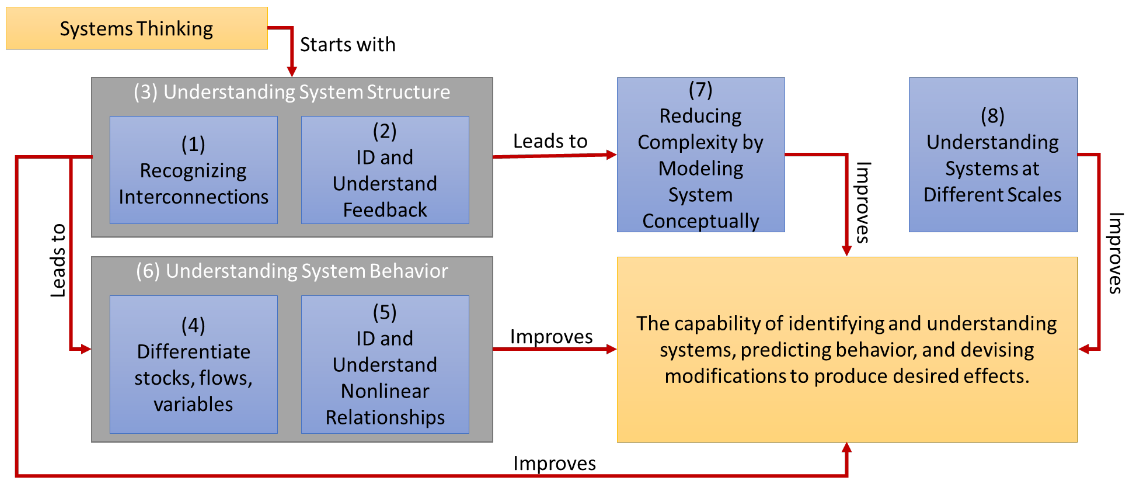


Figure 38: Systems Thinking Systemigram from [67]

3.1.3 *Event Oriented and Feedback Systems Thinking*

Event oriented thinking is a linear action-reaction method of thinking. A problem exists and a solution is presented as a fix; like pop-up targets or putting out fires. This type of thinking is pragmatic, compelling, and extremely common in the military. This line of thinking is ingrained in the Army and even reflects the Army Design Process. While simple and decisive, it can be myopic.[61, 73]

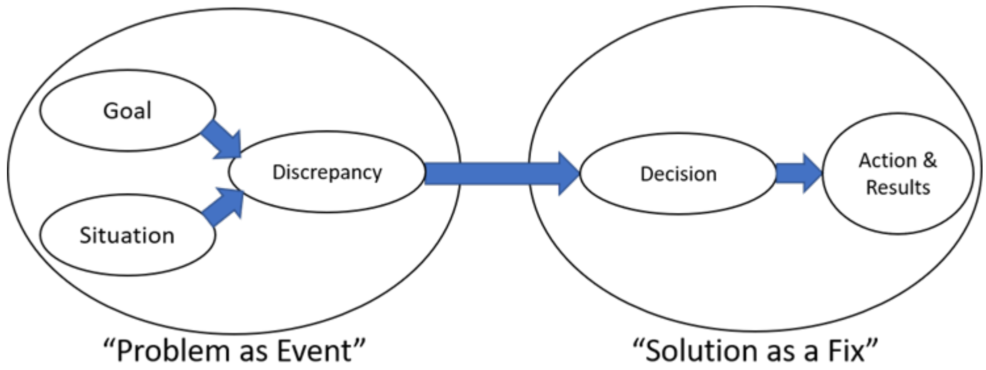


Figure 39 Event Oriented Thinking (recreated from [1])

Figure 39 recreated from Dr. John D.W. Morecroft's book *Strategic Modeling and Business Dynamics: A Feedback Systems Approach*, demonstrates the idea of that problems are often treated as events. When confronted with a discrepancy in each situation with desired goals, leaders decide upon a solution as a fix and see immediate action and results. This method, ideally, provides the desired effects. However, event-driven thinking typically entails short-term solutions and fails to take into consideration the second and third order long-term effects of those decisions.

Event		**Fix**
Increasing Demand	⟶	Deploy more AISR
Intelligence Gaps	⟶	Add new sensors
Too Much Data	⟶	Add More Analysts
Too Many Deployed	⟶	Develop Reach-back
Too Much Data	⟶	Federate PED

Figure 40: Examples of Event-Driven UAS Decisions

As an example of event driven thinking, take the notional example of UAS decision in Figure 40 . Thought treated as independent problems and solutions, the items in the figure are inherently linked.

The feedback systems approach to thinking, on the other hand, is what Dr. Peter Senge refers to as a “shift of mind” that requires “seeing interrelationships rather than linear cause-effect chains” and “seeing processes of change rather than snapshots.”[70] The fundamental premise behind feedback systems thinking is the realization of circular causality. Feedback systems thinking solutions that are arise in response to decisions made in or effects of their organizational and social environment. The fundamental premisis the idea that solutions are not implimented in a vacuum. Problems do not simply stem from events; the both problems and slutions are interdependent.[74]

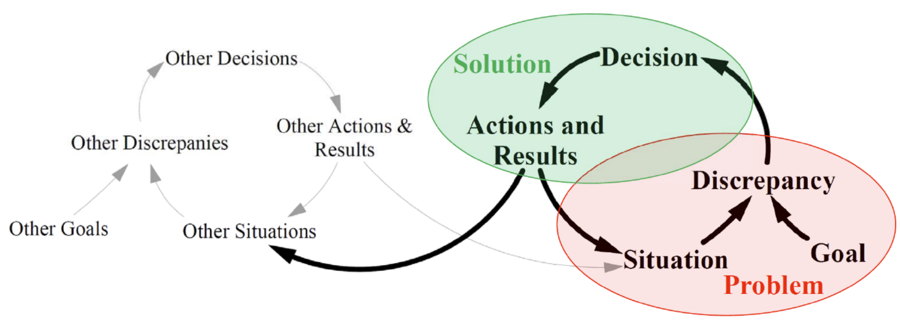


Figure 41: Feedback Systems Thinking Reproduced From[74]

Figure 41 depicts and overview of the feedback systems thinking concepts. A problem exists (in the red circle) where there is a discrepency between the current situation and the desired goal. As a solution (green circle) a dicision is made, actions are taken, and there is a net result. This result impacts the current sitation but also affects other situations, which consequently, through the same cycle have actions and results that feedback and affect the current situation for the problem of interest in a constant loop. This simlified figure shows only a single external loop that is affected by and, subsequently, affects the main problem. However, in reality there can be several, dozens, or hundreds if interlocking loops. Obviously, it can become overwhelming and is beyond human capacity to capture. Therefore, like water seaking the path of least resisitance, humans have a tendancy towards event-driven, linear thining for problem and solutions.

A popular humorous cartoon by Arnie Levin once featured in *The New Yorker Collection* is often used in Systems Engineering as a demonstration of the necessity for feedback thinking and the dangers of event-driven thinking. The danger lies in the concept of hidden feedback. In the cartoon, the first panel depicts a man seated in the center of two large square blocks of stone, one on either side like giant dominoes. From his narrow perspective he has two choices: push over the block on his left or on his right to give himself more space. The second panel of the cartoon showed that his short-term decision was a success; knocking down the block to his immediate left provided him ample breathing room. Like large dominoes, the block he pushed began to topple blocks adjacent to it, confirming to the man that his immediate decisive action was correct. However, in the third and final panel, we see the entire picture and notice that the large blocks of stone are arranged in a giant circle. His decision to knock down the block to his immediate left caused a disastrous chain of events that, despite his best intentions, will ultimately result in the blocks crushing him; demonstrating the risk of not looking at problems holistically.[61]

Too often complex problems, especially in the military, are approached with event-oriented thinking where complex dependencies and non-linear relationships due to feedback loops cannot be readily visualized. Simply increasing one element of the system to meet a particular demand signal (e.g. more unmanned full-motion video (FMV) assets) without realizing its effect on other elements of the complete system and the limitations and delays of those systems over time will fail to render the desired improvements or outcomes; much like optimization of individual parts often fails to optimize the whole.

3.1.4 Tools to Aid System Thinking

3.1.4.1 Sector Maps

For complex systems-of-systems, there are no shortages of potential model activities, elements, and actors. According to Dr. Morecroft "[w]hat matters is not so much the raw number of components but the intricacy with which they are bound together."[61] For the initial analysis of such systems, a "sector map" can be a valuable tool to reduce complexity and identify key actors in the systems, serving as a device for "easing the transition from a mind's eye view" to a complex stock and flow rendering.[71] A sector map is a holistic rendering of a system at a much higher level than causal diagrams or stock-and-flow diagrams. The purpose of these sector maps is to develop simple visual model to spur group discussion of the system and the associated problems while providing a shared understanding of the major components. Once stakeholders are engaged with a concept model, then larger and more refined models and simulators with greater granularity can take shape and be used for analysis, as necessary [61].

Modelers typically use bevel-cornered rectangles, ovals, or circles to represent sectors of the system. These sectors represent 'key actors' in the system rather than quantities or accumulations and normally include the operating policies though stock accumulations within each sector can be added for more detail [71]. Sectors are connected by arrows representing non-dimensionally consistent bundles of flows between sectors. A basic sector diagram for a canonical predator-prey model is depicted below showing the two sectors of the basic system where predators consume prey. The top image in the figure depicts a basic sector map while the lower image depicts a sector map with imbedded stock-flow diagrams that can be developed through stakeholder refinement.

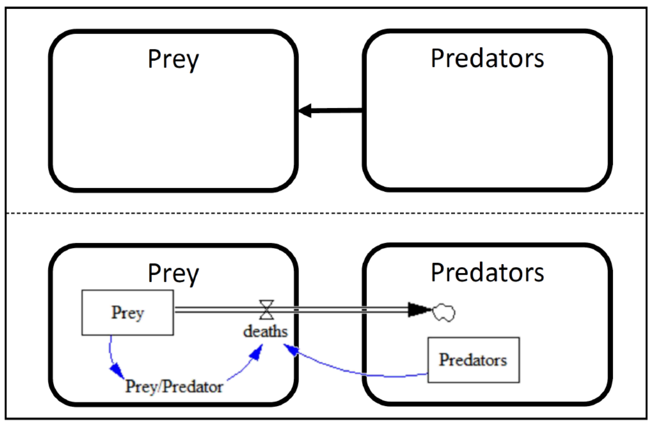


Figure 42: Sector Map Example

3.1.4.2 Causal Loop Diagrams

To develop feedback loops, modelers use systems thinking to first create causal relationships using causal diagrams. These diagrams are comprised of variables as nouns which are positive measurable quantities connected by causal links, represented by arrows. Obviously, causality is required, not simply correlation.

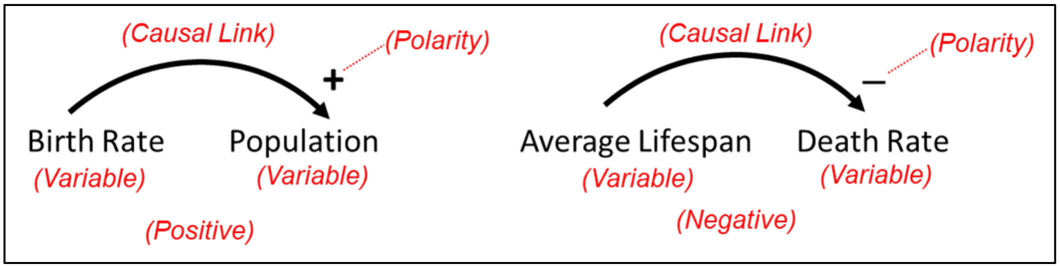


Figure 43: Causal Diagram Examples

The ‘+’ or ‘−’ indicate the link polarity. A positive polarity indicates that if one variable were to increase so to would the other variable, while negative polarity indicates that if one variable were to increase the other would decrease.

Table 6: Causal Diagram Link Summary Adapted from [51]

Polarity	**Definition**	**Math**
 Positive	If X↑ then Y↑ If X↓ then Y↓	$\frac{\partial Y}{\partial X} > 0$
	For Accumulation: X adds to Y	$Y = \int_{t_0}^{t} (X + \cdots)ds + Y_{t_0}$
 Negative	If X↑ then Y↓ If X↓ then Y↑	$\frac{\partial Y}{\partial X} < 0$
	For Accumulation: X subtracts from Y	$Y = \int_{t_0}^{t} -(X + \cdots)ds + Y_{t_0}$

Causal diagrams are a tool to represent feedback that aid the model builder in developing mental models that can be converted into basic algebraic relationships, capturing hypothesis regarding dynamic causes, and communicate the central tenants of the model. Forrester recommends using causal loops as ‘soft’ techniques to explain dynamic behavior to non-technical decision makers, while Sterman and Morecroft endorse using them in the early stages of modeling in parallel with sector maps to be later combined into stock and flow diagrams.[51, 61, 69] When feedback within a causal loop diagram leads to a change in condition that results in a balance effect, it is called a balancing loop and is indicated as a 'B' in a circular arrow. When feedback results in continued increasing or decreasing effect, it is called reinforcing loop and is indicated as an 'R' in a circular arrow. These structure of these feedback loops results in dynamic behavior depicted in

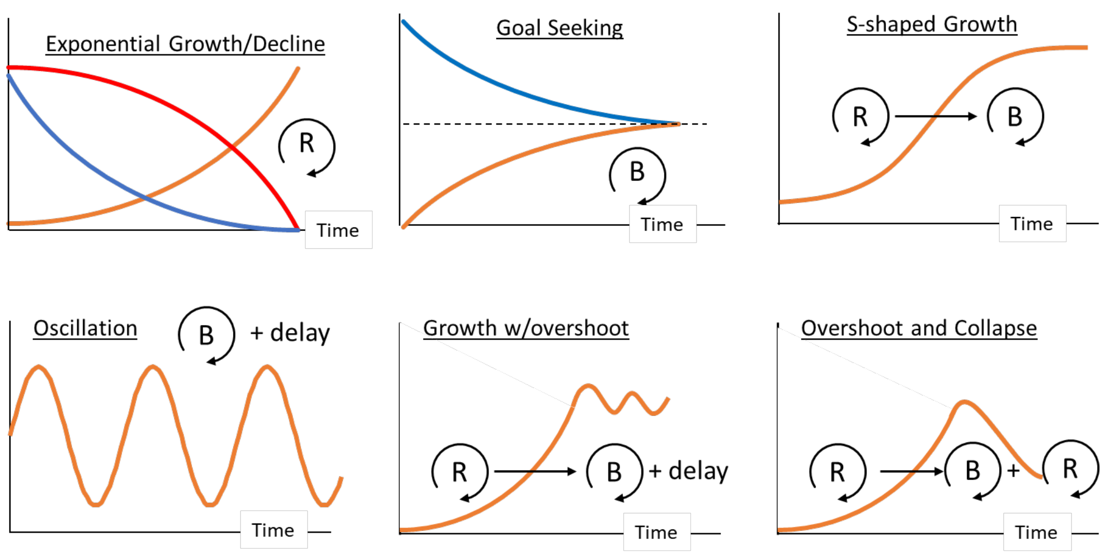


Figure 44: Fundamental Modes of Dynamic Behavior adapted from [61]

3.1.4.3 Integrated Product & Process Development (IPPD)

Created in 1994 at Georgia Institute of Technology by Dr. D. Schrage and Dr. D. Mavris, the Integrated Product and Process Development (IPPD) (Figure 45) is a means to integrating Quality Engineering (QE) methods, System Engineering (SE) methods, computer-integrated environments in a top-down design and decision support process.

This methodology is a procedural approach to design but via "system synthesis through multidisciplinary optimization" also encompasses an analytical approach, and via the "robust design assessment and optimization" is also an experimental approach. This method simultaneously considers the product (e.g. performance, geometry) and process (e.g. manufacturing, economics, supportability) characteristics during design decomposition and re-composition. The overall approach incorporates the early integration and concurrent application of high-fidelity information from all disciplines that play a role in the system's lifecycle into the design stages.

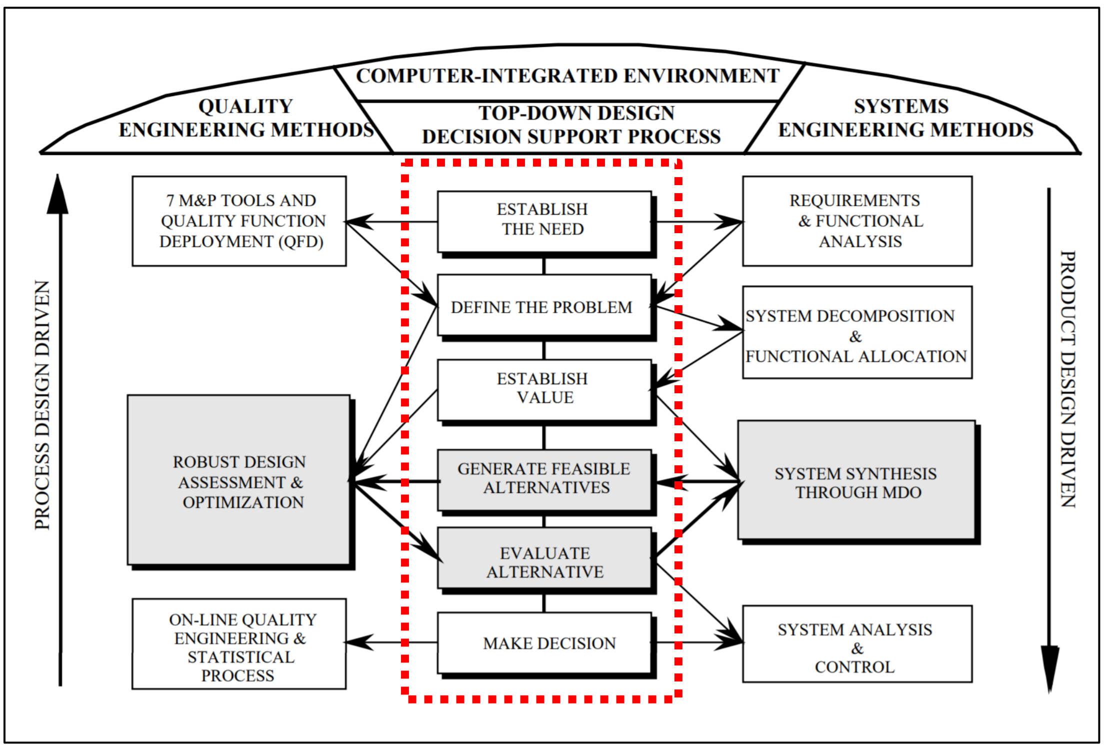


Figure 45: Georgia Tech IPPD Approach Flowcharts from [75]

By integrates product design tools with process design tools in a top down design decision support process this method creates a formal process completed through a computer-integrated environment by which early design decisions can be improved by integrating required knowledge at the onset.[75]

The key take-away from this approach is the center column (outlined in red on the figure) which is the core of the IPPD and the essential steps of a top-down decision process. It is this top-down decision process that will be used as a core of this framework for rapid SoS analysis.

3.2 Modeling and Simulation

To approach complex problems, modeling and simulation are often used to aid human decision-making. This chapter will provide a background on basic modeling and simulation terms to ensure a common understanding of terminology between the author and the readers. To help inform the research questions in subsequent chapters, this section will also provide a brief review the most common types of modeling and simulation in use today and techniques used to create and validate them.

3.2.1 Definitions

3.2.1.1 Models

Depending on the domain of interest, there are many definitions for the word "model." From a review of DOD and Systems Engineering literature the following broad definitions are presented:

From the *DOD Modeling and Simulation (M&S) Glossary*:

> "A physical, mathematical, or otherwise logical representation of a system, entity, phenomenon, or process." [63]

From Dr. Osman Balci's 1994 article "Validation, Verification, and Testing Techniques throughout the Life Cycle of a Simulation Study" in the *Annals of Operations Research*:

> "A model is a representation and an abstraction of anything such as a system, concept, problem, or phenomena." [76]

From system dynamic's expert, Gene Bellinger's webpage "System Thinking":

> "A model is a simplified representation of a system at some particular point in time or space intended to promote understanding of the real system."[77]

From Dr. C.E. Dickerson and Dr. D.N. Mavris's textbook, *Architecture and Principles of Systems Engineering*:

> "A model is a relational structure for which the interpretation of a (logical) sentence in the predicate calculus becomes valid" [29]

With these definitions in mind, a summarized definition is proposed for use in this thesis:

Definition: A **model** is a static, abstract, simplified representation of a system intended for study to the appropriate level of complexity to answer questions and promote a shared understanding

The key element of a model as that it is static. That is not to say that a model cannot be updated or changed, but that its practical purpose is to understand the structure of a system and the interdependencies and relationships rather than the effects of those relationships over time. Models are important when experimenting or prototyping is either physically infeasible or economically nonviable. The process of mapping a real-world problem to a model is called abstraction, which determines the fidelity required in the model.

3.2.1.2 Simulations

Often, the terms 'model' and 'simulation' are used interchangeably, though they are often used in conjunction with one another as a blanket term 'modeling and simulation' (M&S). However, it is important to differentiate the two terms.

From the DOD *Modeling and Simulation (M&S) Glossary*:

> "A method for implementing a model over time." [42]

From Dr. Osman Balci's 1994 article "Validation, Verification, and Testing Techniques throughout the Life Cycle of a Simulation Study" in the *Annals of Operations Research*:

> "Simulation is the process of constructing a model of a system which contains a problem and conducting experiments with the model on a computer for a specific purpose of experimentation to solve the problem."[76]

From system dynamic's expert, Gene Bellinger's webpage "System Thinking."

> "A simulation is the manipulation of a model in such a way that it operates on time or space to compress it, thus enabling one to perceive the

interactions that would not otherwise be apparent because of their separation in time or space.” [77]

With these definitions in mind, a summarized definition is proposed for use in this thesis:

Definition: A **simulation** is the implementation of model over time for the purpose of experimentation

The difference between a model and a simulation is the element of dynamic behavior or the ability to change over time. While a simulation may reach a state of equilibrium it is not initially static unlike a model. Dr. Jerry Banks, in his textbook *Discrete Event System Simulation*, highlights eleven circumstances for which simulation is an appropriate tool and eight rules of when simulation is not appropriate (Table 7).[78]

Table 7: When Simulation Is and Is Not Appropriate

Simulation is Appropriate	Simulation Not Appropriate
To enable the study of and experimentation with complex system internal interactions	If the problem can be solved using common sense.
To simulate the effects of informational, organizational, and environmental changes on a model’s behavior.	If the problem can be solved analytically
When knowledge gained from the design of the simulation would identify and suggest ways to improve the system	If it is easier to perform direct experiments.
When changing inputs and observing outputs can provide valuable insight into the most important variables and variable interactions.	If cost of simulation exceeds potential savings.
When it can be used as an educational device.	If no data or estimates are available to inform the model and simulation
To prepare for future events by experimenting with new designs or policies prior to implementation.	Not enough time or personnel to verify and validate the model.
To verify analytical solutions.	If managers expectations are too high or if abilities of simulation are overestimated.

Table 7 Continued: When Simulation Is and Is Not Appropriate	
To develop requirements by simulating different capabilities for consideration.	If system behavior is too complex it can't be simplified; i.e. human behavior.
As a training device with reduced cost.	
For visualization of an operation	
When a modern system or system-of-systems is so complex that interactions can only be evaluated through simulation.	

3.2.1.3 <u>Characterizations of M&S</u>

M&S has many characterizations that will be used throughout this thesis. A classification chart of these terms is included in Figure 3.13. M&S may be *qualitative* or *quantitative*. In qualitative research, the primary goal is exploration into subjective, nonquantitative or difficult to enumerate characteristics such as opinions, motivations, and structures. By contrast, in quantitative M&S the intent is to generate numerical data or that can be transformed into usable statistics for analysis. These qualitative M&S can be *deterministic* which means they have no random components only known values or *stochastic* meaning it includes random components and must incorporate uncertainty (typically through probability distributions). F

Furthermore, as discussed in the previous sections, M&S are either *static* or *dynamic* depending on if the passage of time is considered. If time is not considered, we use the term model to describe it and simulation if we want to evaluate the effects over time. That is not to say that models cannot be simulated per say. Monte Carlo simulations often only represent a system at a particular point of time but run thousands of cases (typically greater than 10,000) to evaluate the probability of different outcomes in an stochastic processes and account for uncertainty.[78]

Finally, simulations can be *discrete* or *continuous*, like stepwise or continuous mathematical functions, respectively. How the state variables change with respect to time dictate this characterization. Some problems incorporate both continuous and discrete state variables which must be addressed (typically through discretization of continuous variables). [36]

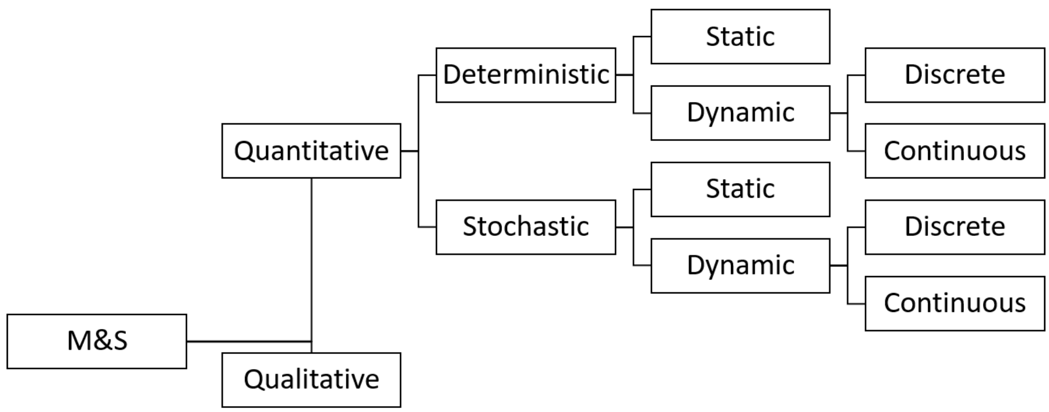


Figure 46: M&S Characterizations

3.2.2 Model Development Techniques

3.2.2.1 <u>Forrester's Method</u>

To develop models, Dr. Jay Forrester [69]recommended the six-step process depicted in Figure 47 in the year 1994. As shown, the process developed for System Dynamics but could be easily applied to any M&S paradigm. Like any modeling process, Forrester's is iterative.

The first step involves describing the system. This can be accomplished through a variety of means, to include case studies, soft operations research, and applying systems thinking. The intent of this step is to identify behavior of a system or SoS that is undesirable with the aim of improving it. The second step involves formulation of the M&S in terms of level and rate equations with more explicit information. The third step is the actual simulation of the model and check for unrealistic behavior, cycling back to Steps 1 and 2 as necessary to check for common errors such as simultaneous equations, terms defined more than once, and inconsistent units. This cycle repeats until the behavior is adequately representative of the system under consideration. The fourth step involves testing policy alternatives to identify policies that indicate the most promise to reaching desirable behavior. Many of these policies are proposed by SMEs. The fifth step involves gaining consensus for the selected implementation of policies through exploration, gaming, and debate with key stakeholders. The final step involves the implementation of policies generated in the M&S from the previous steps. Like the preceding steps, the M&S is considered a 'living' environment that can continually be refined and adjusted as real-world observations of behavior confirm or deny the M&S behavioral predictions.

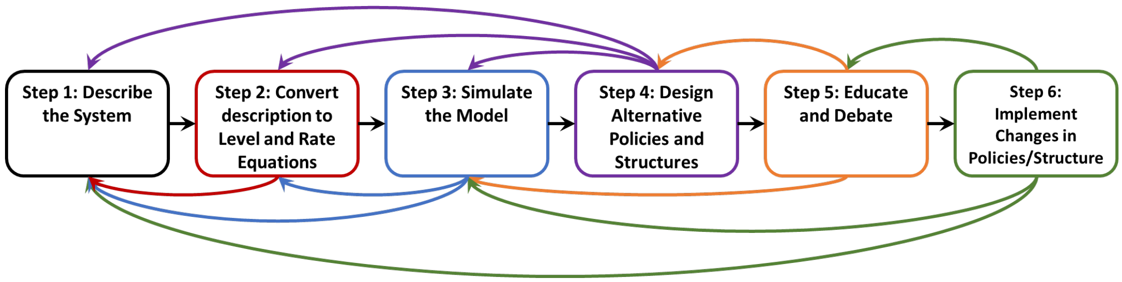


Figure 47: Forrester's 6-Step SD Modeling Process recreated from [69]

3.2.2.2 Sterman's Method

From his book *Business Dynamics: Systems Thinking and Modeling for a Complex World*, Sterman's modeling process and its five steps are summarized briefly in Figure 48 [51].

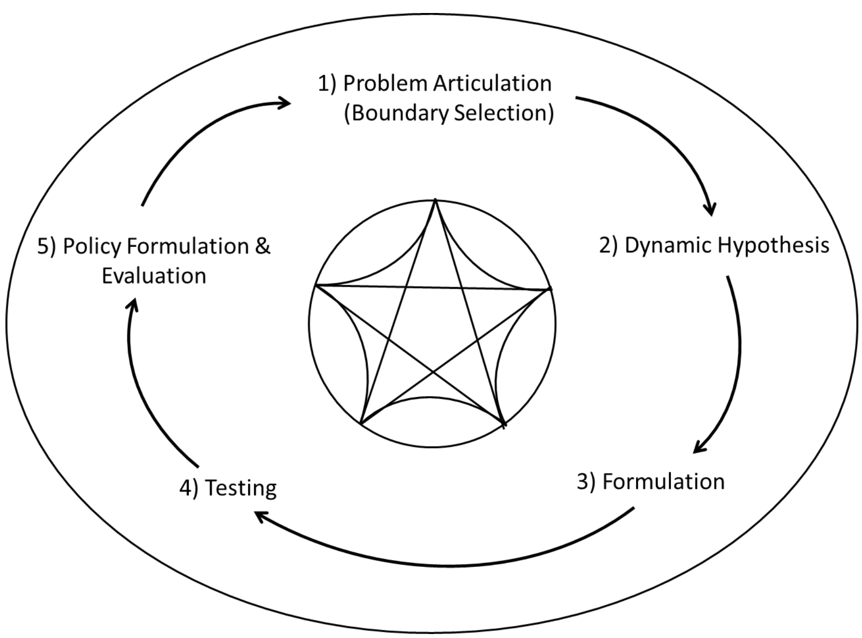


Figure 48: Sterman's Iterative Modeling Process recreated from [51]

Sterman's process that he published in 2000 varies slightly from that Forrester devised in 1994. Like Forrester's methodology, it is an iterative process and not a linear sequence, though he emphasizes that defining the problem is the most important step in the process.

Table 8: Steps of Sterman's Iterative Modeling Process adapted from [34]

Step 1: Problem Articulation	Define the problem dynamically and why it is a problem. Establish *reference modes* and *time horizon. Reference modes* are descriptive data and graphs showing the growth of the problem over time. *Time horizon* is how far back the problem arose.
Step 2: Formulate a Dynamic Hypothesis	Characterize problem in terms of feedback (causal loops) and stock and flow structure of the system. Additional tools include mapping techniques such as subsystem diagrams, policy structure diagrams, and model boundary diagram.
Step 3: Formulate a Simulation Model	Create computer simulation by transforming dynamic hypothesis int detailed diagram of feedback processes and associated algebraic equations.
Step 4: Testing	Begins as soon as first equation is written but more than simple replication of historical behavior. Conduct sensitivity analysis considering parametric and structural assumptions. Must test under extreme conditions.
Step 5: Policy Design and Evaluation	Design and evaluate policies for improvement; conduct "what if" analysis. Don't simply change parameter values but change structure, decision rules, and delays.

Sterman recommends and describes twelve procedures for evaluating the acceptability of a model: boundary adequacy assessment, structure assessment, dimensional consistency assessment, parameter assessment, extreme conditions tests, integration error tests, behavior reproduction testing, behavior anomaly tests, family member tests, surprise behavior tests, sensitivity analysis, and system improvement tests.[51]

3.2.2.3 Verification and Validation (V&V):

The terms ‘verification’ and ‘validation’ (V&V) generally have commonly agreed upon definitions in engineering, acquisitions, and in modeling and simulation. However, even very experienced practitioners have a tendency to conflate or confuse the terms.[79]

Model verification is the process of evaluating if a model is transformed from one form into another, correctly and completely during a given development phase to satisfy the conditions imposed at the start of that phase.[76, 80, 81] In other words, verification ensures that the model is behaving the way it was intended and answers the question “Did you build it right?” [64, 79].

Definition: Verification is the act of ensuring that a model is built and functioning correctly. Is it built right?

In order for the verification process to be successful, the model must include all of the components specified during system definition and must run without errors or warnings [79].

Validation, on the other hand, is the process of determining if a model or simulation accurately represent the real world system it meant to represent within its domain of applicability from the perspective of the intended study objectives of the model.[63, 76] In other words, to validate a model is to determine if the model behaves the same as the real system and answers the question, “Did you build the right thing?" [64, 79].

Definition: Validation is the act of ensuring that a model is correct for the system meant to be represented. Was the right model built?

V&V from a systems perspective is often visualized with the classic 'Vee' diagram shown in Figure 49. It starts from the left side of the 'Vee' beginning with user requirements and moves down through the decomposition and definition process until reaching fabrication and moving back up the right side of the 'Vee.' At each step there is verification that the elements are built correctly culminating in a validation that the system meets user requirements. While the model mentions 'code' at the base of the 'Vee', this method is mostly applicable to physical systems.

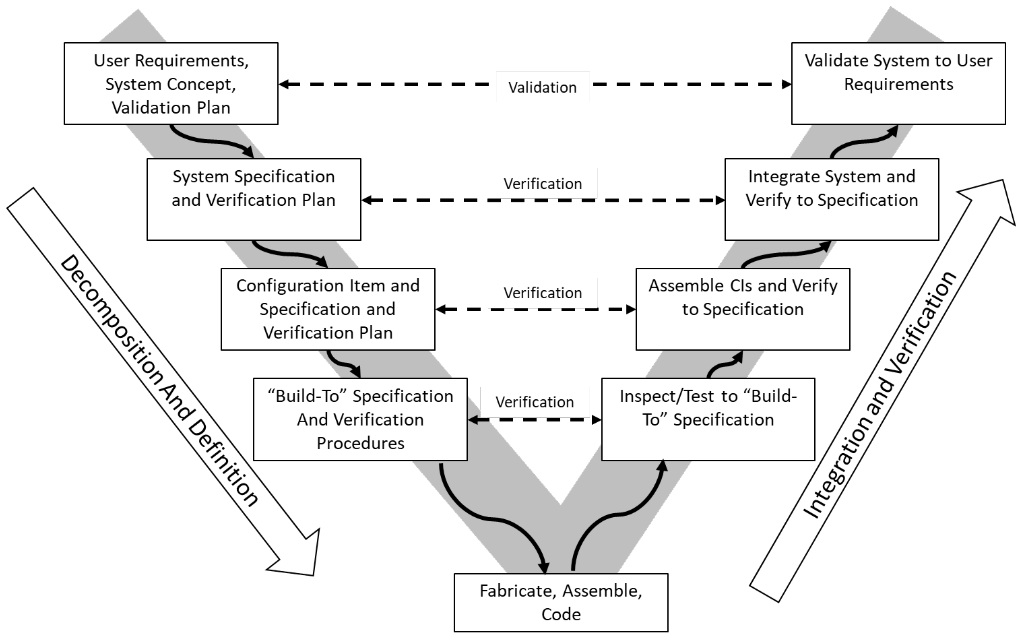


Figure 49 'Vee' Model recreated from [29]

A method for verifying and validating the model and simulation for the proposed research will be discussed in greater detail within the experimental plan.

3.2.2.4 <u>Model Errors</u>

Like any experiment, model errors are a significant concern. A model that does not accurately represent the system under study then the results are not reliable. There are three types of model error and these types of model error are analogous the errors found in inferential statistics. The premise of inferential statistics, like V&V, is not to prove the null hypothesis (H_0), but to attempt to disprove it by finding an exception (H_1); much like was discussed as the purpose of V&V in the previous section. By failing to prove the model wrong during V&V under the conditions and for the problem the model was built, the confidence in the model's predictive abilities is increased. A commonly used graphic in statistics is shown in Figure 50 to demonstrate the first two types of errors associated with this idea.

		Null Hypothesis (H_0) is:	
		True	**False**
Decision about Null Hypothesis (H_0)	**Reject**	**Type I Error** "False Positive" (Probability = α)	**Correct Inference** True Positive (Probability = $1 - \beta$)
	Accept	**Correct Inference** True Negative (Probability = $1 - \alpha$)	**Type II Error** "False Negative" (Probability = β)

Figure 50 Statistical Error Types

In statistics, Type I Error is rejecting the null hypothesis when it is true. Likewise, in modeling and simulation, this equates to rejecting model credibility when it is in fact sufficiently credible (H_0=model is credible) and is referred to as "Model Builder's Risk." In statistics, Type II Error is accepting the null hypothesis when it is false. Likewise, in modeling and simulation, this equates to accepting the model as credible when it is, in fact, not sufficiently credible; this is known as "Model User's Risk." The last type of error of interest is Type III Error. In statistics, this is known as correctly rejecting the null hypothesis, but for the wrong reason. In modeling and simulation, this equates to solving the wrong problem with the model that was created [76].

3.2.2.5 Variability and Uncertainty

In any model, uncertainty will be present through either data, the process, or assumptions made in the model. Two types of uncertainty are relevant: aleatory variability and epistemic uncertainty, also referred to as randomness and uncertainty respectively.

> **Definition: Epistemic uncertainty** is the uncertainty regarding the model due to limited data and knowledge.

If variables are discrete and random, the epistemic uncertainty can be modeled by probability distributions. If the variables are random and continuous, probability density functions can be used to model epistemic uncertainty. Epistemic uncertainty is often confused with aleatory variability.

> **Definition: Aleatory variability** is the natural randomness in a process.

A method to distinguish between epistemic uncertainty and aleatory variability is to examine the parameter in question. If the parameter has different values and different times randomly, then it is aleatory variability. If the parameter has either one value or another value, but the modeler is uncertain as to which value it has then the parameter has epistemic uncertainty [82].

3.3 Modeling and Simulation Paradigms

3.3.1 Petri Nets

3.3.1.1 Description

The original Petri Nets were presented by Carl Petri in his 1960's PhD Thesis.[83] Commonly used for data flow, basic Petri Nets, or place/transition (PT) nets are a formal, graphical, executable technique for the specification and the analysis of concurrent, discrete-event dynamic systems. Because Petri Nets having multiple transition pathways between states, they are a non-deterministic modeling tool. This means that multiple executions of the same model are not guaranteed to have to same results [84].

3.3.1.2 General Principles

Petri Nets are directed bipartite graphs with two disjoint node types and are a type of simulation under the category of Discrete Event Dynamic Systems (DEDS). Petri nets consist of places (circles) and transitional nodes (rectangles or squares) joined by arcs. Arcs may have multiplicities. Places typically represent conditions and while transitional nodes represent events that may occur. An indistinguishable non-negative integer number of markers, known as tokens, are added to the place nodes and the distribution of tokens in the nodes (known as a marking) fully indicates the state of the system. Based on transition rules or conditions requiring every one of the input places to have at least as many tokens as the multiplicity of the arc connecting the place node to the transition to 'fire'.[30] The marking will enable transitions that can then 'fire' and remove a number of tokens from place nodes (based upon output arc multiplicity) and create corresponding tokens in other place nodes (based upon input arc multiplicity). If tokens are on a place node then the place exists; meaning more amount or activity of a given entity it represents in the model exists.

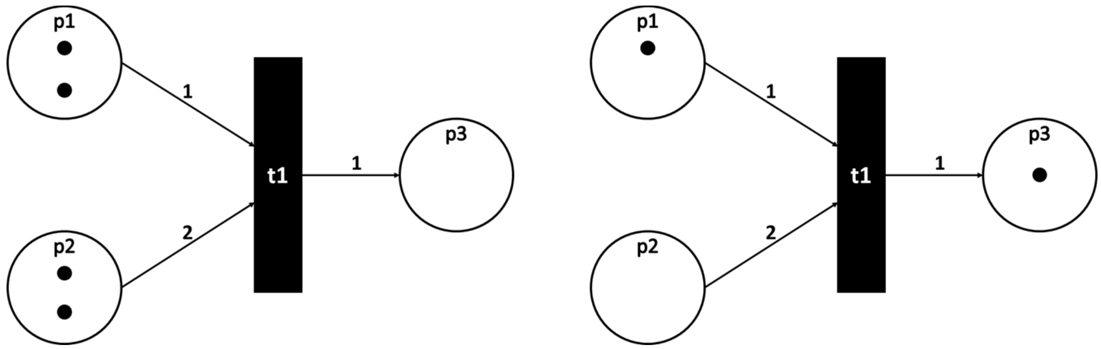


Figure 51 Basic Petri Net and Example Transition

Transition firings represent any discrete event of the system being modeled.[84] The corresponding moving of the tokens around based on the topology of the graph represent changes in state of the system. Of note, Basic Petri Nets are not suitable for representing systems typified by time-driven processes like those described by differential or difference equations as they did not account for the concept of time [30]. Rather, all enabled transitions would immediately fire. Later developments in Petri Nets allow for deterministic time delays called timed Petri Nets [84]. This lead to the creation of Stochastic Petri Nets (SPN) where delays are assigned as random variables placed on the node based on given distributions to account for uncertainty [85]. Because of the non-deterministic nature of the model, Monte Carlo Simulation can be performed to account for probabilistic outputs to account for these uncertainties.

Another variation of the method is the colored Petri Net which uses colors to differentiate between tokens. The colors can be used to track specific tokens or be leveraged as a flag to account for token which have passed through a specific transition or have demonstrated a specific behavior. The colored Petri Nets also provide the benefit of reducing the size and number of states required to create the model. [57]

The final variation of note in the application of aging tokens. In this variation, the ages of tokens are tracked as they transition through the place and transition nodes in the system. Transition nodes can be assigned rules that only accept tokens within specified age requirements [86].

3.3.1.3 Examples of Use

Under the precursor to DoDAF, the C4ISR architecture, the default paradigm for an executable model was DEDS. Command and Control information systems that consist of primarily of computers and software are intrinsically discrete event systems in terms of data flow, making Petri Nets a logical and appropriate paradigm [30]. Dr. Levis, himself, demonstrated the ability to evaluate distributed C4ISR architecture utilizing colored Petri Nets in 1991 [87]. However, while this may be true for analyzing the flow of data, it fails to capture the other present and contributing elements of the system-of-systems, to include but not limited to, experience, training, enemy actions, and kinetic effects.

In version 1.5 of the DoDAF, the use of Petri Nets for the refinement and dynamic analysis of Operational Viewpoints, namely the Operational Rules Model (OV-6a) Operational State Transition Description (OV-6b), and the Operational Event-Trace Description (OV-6C). It additionally identifies Colored Petri Nets as valuable tools for the dynamic simulation of the OV-5, Operational Activity Model. While the newest *DOD Architecture Framework Version 2.02, Change 1 Architect's Guide* .makes no mention of Petri Nets and provides no guidance on executable techniques, the method remains a valid approach to dynamically assessing discrete information flow systems [88, 89].

Petri Nets using aging tokens have been used to model systems that have components that may degrade with time like the tokens that represent them. Hence, Petri Nets with aging tokens have been widely used for reliability, maintenance, and safety studies. This property also makes the useful for information systems that contain information that becomes irrelevant after a certain period. AISR PED architecture is one such example where intelligence products become less valuable over time. In military terms, this is when certain priority intelligence requirements pass the LTOIV (latest time information of value) cutoff. Petri Nets with aging tokens can be used as a valuable tool to identify choke points that stagnate information flow and cause it to expire before being of use [86].

3.3.2 Discrete Event Simulation (DES)

3.3.2.1 <u>Description</u>

Discrete Event Simulation (DES) traces its roots to the General-Purpose Simulation System (GPSS) created by Geoffrey Gordon of IBM in the early 1960s and is a type of global entity processing algorithm, where the elements may be stochastic or deterministic. In a system or a process where the state variables that define the system change only at discrete points in time, much like a step function in mathematics, DES may be appropriate. For simple systems over short periods of time such as a simple single-channel queue (such as a bank teller or market cashier), analysis can be computed analytically, typically by hand or with a spreadsheet.(Figure 52) However, DES is necessary when multiple dependent and interactive events occur, often on parallel timelines, simultaneously. However, unlike analytical models, DES is a numerical method to 'run' rather than to solve mathematical models [78]. The passage of time plays a crucial role in discrete-event simulations, and hence they are dynamic.

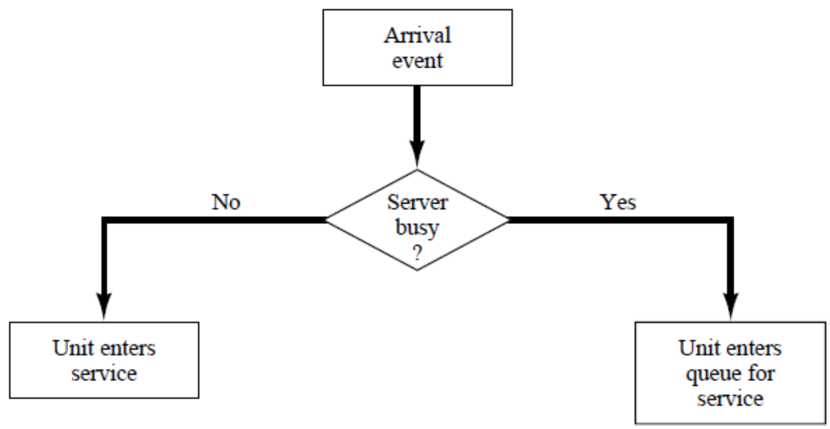


Figure 52: Simple Unit Entering System Flow Diagram recreated from [78]

In DES systems are typically characterized by flow charts comprised of entities (passive objects such as parts, people, documents, etc.) and push these entities through blocks (sometimes referred to as servers [90] [91]. DES model a system as a network of activities and queues, where each entity is individually represented and can be tracked through the system. Each of these entities is assigned specific attributes to determine what happens to them through the duration of the simulation [92].

When modeling a DES, three primary paradigms exist: Activity Oriented, Event-Oriented, and Process Oriented. Using the example of simple queue where customers arrive to a single teller/server where each service performed has a varying random duration. Subsequent customers must wait in the queue until the previous customer is served. In this system, time is continuous, both for the random service time variables and for the arrival times. The paradigms explain how this problem will be made discrete.

The activity-oriented paradigm is the original and most basic DES paradigm. In this method, activity time is segmented into in very small increments. At each time-step, the DES would check for the possible occurrence of specified events. Obviously if the discretization is too small relative to the activity duration, the simulation can be very slow to execute with little to no state change to the system at each step. Depending on the scale and complexity, these simulations can take days to run and be computational expensive. Therefore, “what-ifs” and trade space exploration even with a proper design of experiments can take a very long time and be computationally expensive with a lot of wasted processor time. However, if the discretized time steps are too large, events will be missed.

To address these issues, the event-oriented paradigm for DES was developed. By skipping periods of inactivity and advancing directly to the time of the next event, updating all entity attributes accordingly, processing time is reduced dramatically. To accomplish this task, an ‘event set’ is created which stores all the pending events such as the next arrival in the queue usually in a linearly linked list called a future event list (FEL). This list contains all future scheduled events as event notices. Once the event is drawn, its duration is selected from a sample from a statistical distribution or is computed. To account for real-world unscheduled random events, the events are “represented by the end of some activity” as a statistical distribution [78]. The FEL is ordered chronologically by event time(t_n) and the clock time (t) where the clock time progresses to and executes each event on the FEL.

$$t < t_1 \leq t_2 \leq t_3 \ldots \leq t_n \tag{1}$$

The contents and lengths of the FEL changes as the simulation progresses: proper list processing will increase the efficiency of the computer program. New events are added to the list in chronological order based on their future event time (t^*) by taking the current clock time and adding the activity time(a) for the new event (calculated or generated from probability distribution) and comparing t^* to the t_n times of the other events on the list. This method is called *bootstrapping*. This process continues until the predesignated run time T_E. Such a method is only possible because of the discrete nature of the system and would not be possible to implement in a continuous simulation model. This paradigm is easy to implement, exhibits fast execution speed, and is flexible.

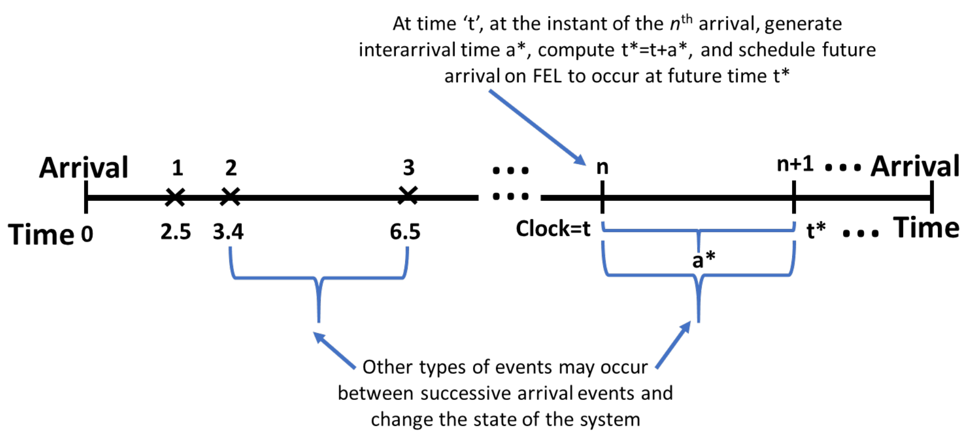


Figure 53: DES Bootstrapping adapted from [78]

The final DES paradigm is called process-oriented DES. As the name implies, activities are modeled as a process, with each process modeled as an independent thread (i.e. one thread for the customer arrivals and one thread simulating the teller, and a general thread to manage the global process). While legacy software made this type of DES difficult, the introduction of more user-friendly thread packages such as Unix, SimPy, and

Java, to name a few, has made this method of programming DES the most popular. The code is more modular and is easier to read and write than other DES paradigms [91].

3.3.2.2 General Principles of DES

DES is one of the most popular M&S paradigms for good reason; many real-world processes in logistics and manufacturing can be modeled as a network of queues and activities. DES excels at providing insight on processes concerned with individual entities at a high level of fidelity and low aggregation (Figure 80) by allowing users to "explore progression through a system" [93]. It is ideal for exploring the impact of randomness analyzing system behavior at for operational/process view and has been used in conjunction with other operations research methods such as statistical analysis, data mining and multi criteria decision-making. Because of the aleatory variability of normal flow operations that can be difficult to analyze manually, DES is ideal for its incorporation of and emphasis on stochastics. However, DES is primarily concerned with process details and event focus and state changes, not emergent behavior and does not overtly seek to model feedback loops. DES is essentially a model of linear relationships and lacks feedback systems thinking. While this is adequate depending on the goals of a study of a system (creating statistical observations of system such as average delays, average items manufactured, variance, etc.) its aim is to analyze the outputs of a system and explore the impacts of randomness and system behavior and not system structure itself [74, 93, 94].

3.3.2.3 Examples of Use

DES has been in use for so long, it would be impossible to list all its uses. It is best for dynamic process that are essentially a network of queues and activities with multiple entities and stochastic state variables. *Use Cases of Discrete Event Simulation*, a collection of articles from modelers and experts from ten different countries demonstrates the vast applicability of DES across various domains. Examples include using DES to investigate the effectiveness of variance reduction techniques in manufacturing, call centers, and warehouse cross-docking distribution systems; DES of energy consumption in automotive industries; and planning earthwork processes using DES, to name a few [95].

Military acquisitions and operations researchers are heavy users of discrete event simulation to gain insight into a myriad of issues that are wrought with uncertainty but vital to training, maintaining and sustaining a "viable military industrial complex" [36]. In recent military applications it has successfully been used simulate and study military deployment operations [96], assess military helicopter maintenance and the implementation of proposed maintenance free operations [97], military aircraft sustainment [98], and Marine Corps unmanned logistics systems (ULS) concepts of operation [99].

3.3.3 Agent Based Modeling (ABM)

3.3.3.1 <u>Description</u>

Agent-based modeling (ABM) draws its roots from game theory, complex system theory, artificial intelligence, and other disciplines. Though conceived far earlier, ABM was not in widespread use until the early-mid 1990s due the lack of advanced computing ability and the advent of object-oriented programming and implementation [90, 100]. In the early 2000s the use of agent based modeling rose dramatically with widespread used in both civilian and military modeling efforts [35, 101].

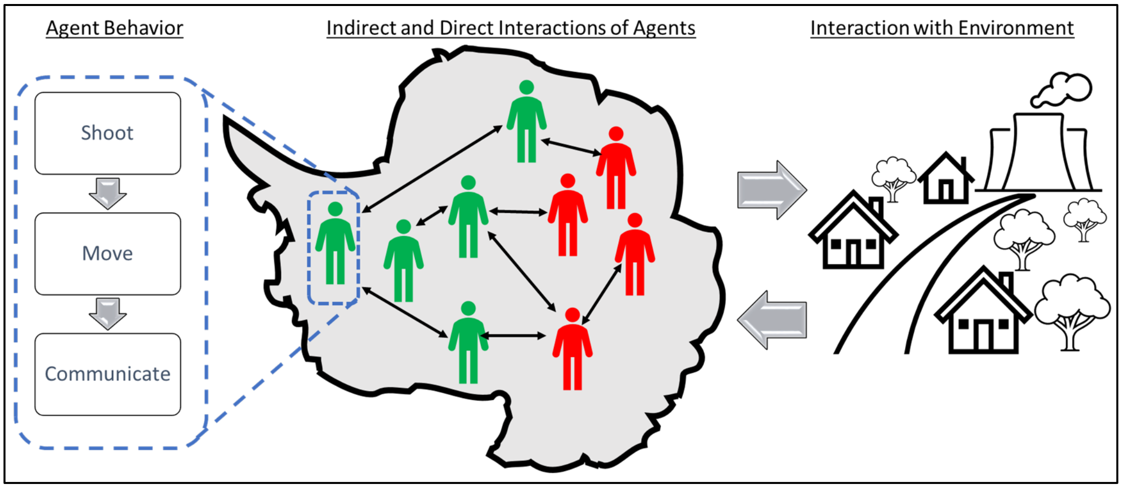


Figure 54: ABM Generic Architecture adapted from[102]

ABM is essentially bottom-up modeling, which Dr. Eric Bonabeau, founder of ICOSYSTEM, describes as a mindset rather than a technology; a mindset of "describing a system from the perspective of its constituent units" [101]. Rather than modeling a macro system with high aggregation, which requires an understanding of overarching structures, mathematical differential relationships, and predicted behaviors, ABM is constructed at micro-level, regardless of the overall level of use for the model. Entities within the simulations are represented by autonomous individual agents with programmed decision-making behaviors based on a set of programmed rules. What distinguishes ABM from other M&S paradigms is that it is decentralized; the interactions of the agents is governed by the agents themselves and not by the system structure [36]. Because of this fact, the model can be constructed without *a priori* knowledge regarding global interdependencies. The repeated competing interactions and relationships between the agents and the environment generates the complex dynamics of the model over time; far beyond what is possible with purely mathematical methods [90, 101].

The complex behavior patterns are the result of aggregate behavior of the agents as a collective and provide valuable insight into the dynamics of the real-world system it is attempting to represent. The consequences of these interactions and the agents' decisions have both spatial and temporal elements. In fact, the ability to incorporate spatial as well as probabilistic aspects of the system are two of the largest advantages of ABM over other flow chart based simulation paradigms [100, 103, 104]. These agent interactions are predominantly nonlinear, meaning the overall collective behavior can only be determined by the interactions of the isolated components, not by simply summing the behavior of them.[105] In other words, simply inspecting the laws satisfied by the components does not reveal regularities in the system behavior. This phenomenon is what is known as emergent behavior. The *DOD Modeling and Simulation Glossary* notes that:

> "If emergence happens over disparate size scales, then the reason is usually a causal relation across different scales. In other words, there is often a form of top-down feedback in systems with emergent properties. These are two of the major reasons why emergent behavior occurs: intricate causal relations across different scales and feedback."

This emergent behavior the driving benefit of ABM, though Bonabeau identifies two other benefits; AMB provides a natural description of the system and it is flexible [101]. Emergent behavior has the potential to be counterintuitive as simple changes to the rules that dictate individual agent behavior can have a large effect on the collective emergent behavior. This is a large benefit. By modifying the environmental parameters and the rules the agents follow, researchers can attempt to better understand what the outcomes will be as a result of the emergent behavior of the overall system [105]. This ability to generate variation of simulated data directly from carefully specified set of rules rather than direct measurement of the real world can invaluable when experimentation or measurement is impossible, difficult, or expensive. The simulated data can then be analyzed inductively through observation and generalization [63]. The rules for the agents need not be fixed; more advanced individual behaviors such as learning and adaptations can be generated using neural nets, machine learning, and other modeling types incorporated within the ABM to make adaptive agents [100, 101]. Because ABM generates emergent behavior, it has the potential to change the scientific process from the quest to explain observed phenomenon/behavior to a quest to grow the desired behavior from the constituent entities [101].

To define global interdependencies in terms of governing mathematical equations can be a difficult proposition in some cases. It is far easier to describe how individuals or entities within a system will conduct activities. This is what is meant by a *natural description of a system*—one that is far easier to explain and justify when developing and presenting the M&S to the decision makers or a non-technical audience to aid in building trust for not only the M&S but of its results and conclusion derived therein.

In terms of *flexibility*, ABM can span levels of description and aggregation depending on the problem and the problems under study, as illustrated in Figure 80. ABM is especially useful in problems where the proper level of description or complexity is not known *a priori*. In such a case, agents can be added or removed as necessary given some experimentation with the model. In addition, the behavior of agents, rules of interaction, and ability of agents to think, learn, and evolve (degree of rationality) can all be modified with relative ease; allowing one to play games and 'tune' the model. However, the ability to easily use and program ABM can lead to the misconception that the concepts are easy to master. Bonabeau [101] highlights five examples of when the use of ABM is appropriate, which are summarized and paraphrased below:

1. When agent interactions are complex, nonlinear, discontinuous, or discrete and the behavior of agents can dramatically affect the behavior of others.
2. When then environmental space is central to the problem definition, assumptions, or constraints (traffic, market, etc.)
3. When the populations under study are heterogenous.
4. When agent interactions between agents are complex and heterogenous; they cannot be summarized in a homogeneous global manner.
5. When agents can learn and adapt rather than prescribe to fixed behavior.

As is implied, ABM is not the quintessential M&S paradigm; the model must serve the tailored purpose for which it is prescribed; no general-use model is possible. Soft factors that are wrought with uncertainty such as irrational human agents and subjective choices can be difficult to verify and validate. When such factors are included in a model, it is imperative not to conflate qualitative outcomes as purely quantitative results. The largest challenge in ABM is scale. As previously mentioned, ABM can be used at nearly every level of aggregation. However, in practice, models can easily grow so large that they become computationally expensive and infeasible. Therefore large systems prove to be a challenge for ABM; with large computational time required, making numerous means and ways trades is time consuming, difficult, or impossible without an appropriate design of experiment and meta-modeling, which can further obfuscate the validation of the model and the traceability of root causes and/or compound errors depending on model fit.[106]

3.3.3.2 Examples of Use

Bonabeau [101] also classified four primary classification areas for which ABM and its ability to demonstrate emergent phenomenon and global behaviors: flows (traffic, customer, etc.), markets (stocks, strategic simulation, etc.), organizations (operational risk), and diffusion (innovation and adoption dynamics) [101]. However, since he authored his seminal colloquium paper on the topic, the use of ABM has exploded. It is now used in biology, business, network theory, economic and social studies, and organizational studies. ABM has also seen heavy use in the military since the early 2000s and is now, along with DES the primary means of modeling beyond the engineering level. In fact, the DOD uses ABM to "model everything from maintenance processes on the flight-line or on-board a Naval vessel to examining personnel retention" [35]. ABM has also been used to research and examine historical battles to validate historical observations and assumptions with respect to causes of successes and failures, as well as to use historical battles to validate current models used to predict future battle/engagement results [107].

3.3.4 System Dynamics (SD)

3.3.4.1 Description

System Dynamics (SD) is one of the oldest of the M&S pedagogies examined in this thesis. Jay W. Forrester of the Massachusetts Institute of Technology developed the approach in the mid-1950s and published the seminal book in the field, *Industrial*

Dynamics [108] in 1961. Forrester conceived the fundamental idea for system dynamics while attempting to solve a business problem for General Electric. General Electric could not understand why they had an oscillatory unstable workforce requirement (having a few years of working triple shifts to satisfy demand followed by having to lay off half of the work force). From his modeling of the hiring and operational practices, Forrester discovered that "even if incoming orders remained constant, employment instability could still arise as a consequence of common decision-making policies."[109] His findings became known as the "Forrester effect" or bullwhip effect wherein customer demand results in increase inventory swings with decreasing forecast accuracy as one moves further up a supply chain. Hence, system dynamics was born of business and management to assess trends associated with policy, decision-making and strategy. Modelers would study a corporation, develop a model, and provide consultant services and recommendations. Since its inception, however, SD use has significantly expanded and evolved. The System Dynamics Society's modern definition of System Dynamics is as follows:

> "System Dynamics is a computer-aided approach to policy analysis and design. It applies to dynamic problems arising in complex social, managerial, economic, or ecological systems—literally any dynamic system characterized by interdependence, mutual interaction, information feedback, and circular causality." [110]

The key takeaways from the definition that reflect the core principles of system dynamics are as follows: complexity, interdependence, interactions, circular causality, and feedback. Unlike DES, the key to system dynamics is the idea that behaviors of the model result from nonlinear relationships and feedback. Central tenet of SD being that it is the structure of a system that causes its behavior endogenously, whereas exogenous factors are, at most, triggers of system behavior but not the cause. As a method, system dynamics is

> "a perspective and set of conceptual tools that enable us to understand the structure and dynamics of complex systems. System Dynamics is also a rigorous modeling method that enables us to build formal computer simulations of complex systems and use them to design more effective policies and organizations." [51]

Since Forrester was originally an electrical engineer, SD modeling traces its roots to control theory extended from complex machines to complex systems and thus, draws many parallels. In a closed-loop control system, a portion of an output signal is fed back to the input whereby actual output can be compared to desired output and adjustments can be made to account for perturbations or to reduce error (Figure 55) [111].

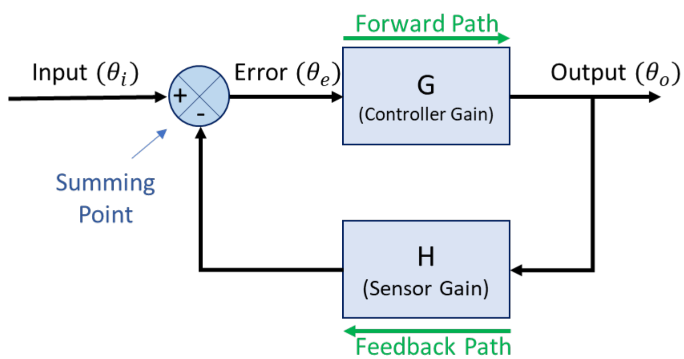


Figure 55: Canonical Closed-Loop Control System adapted from [111]

Hence, concepts such as positive and negative feedback, sensitivity to initial conditions, and state space as well as oscillations, asymptotic behavior, exponential growth, S-shaped growth, goal seeking behavior, overshoots and collapse are as relevant in SD as they are control theory [51, 61]. Developing an SD model requires that one describe processes of accumulation and feedback. These models are then used to systematically test proposed policies for achieving desired outcomes [94].

In SD, real-world systems and processes are represented in terms of stocks, flows, delays, and the information and relationships that determine the values of the flows (Figure 56). A *stock* is an accumulation that characterizes the state of the system under study. Thus, in modeling terms, stocks are the state variables and in mathematical terms represent the integral value or levels. Flows represent the time rate of change in the amount of stock flowing into or out of the system; hence flows are the rates or derivative functions. Additional detail can be added to the logic of stock and flow diagram via *converters* (or *auxiliaries*) that affect the flow rates due to causal relationships. The numerical relationships between the variables is called a *connector*. [51, 61]

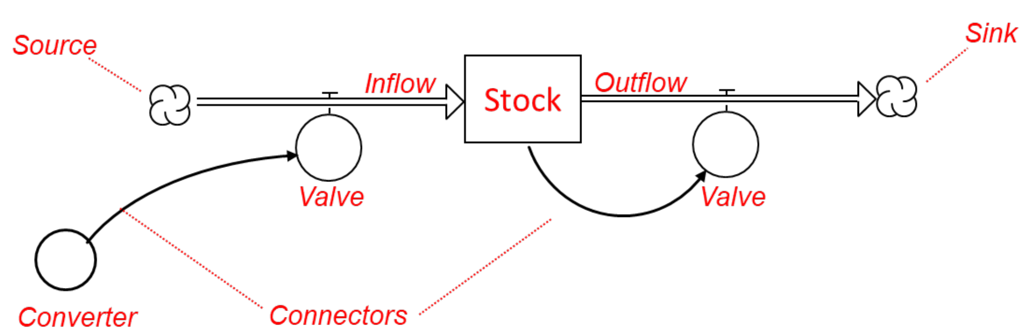


Figure 56: SD Canonical Stock and Flow Diagram

In mathematical terms, the premise of stock and flow is analogous to Reynolds Transport Theorem (RTT) for a control volume; the Eulerian approach used in thermal-fluid systems as mass flows in and out of a system. RTT theorem states that for an extensive property (B) which depends on the mass or size of a system, that the rate of change of (B) in the system is equal to the rate of change of (B) into the control volume plus the net amount of (B) exiting the control volume across the control surface with mass flow. [112]

$$\frac{dB_{sys}}{dt} = \frac{d}{dt}\iiint \rho b dV + \iint \rho b \mathbb{V} \cdot d\mathbb{A} \tag{2}$$

$\boldsymbol{\rho}$ ***is density***; $\boldsymbol{b}$: ***intrinsic property***, $\boldsymbol{V}$: ***volume***,

$\mathbb{V}$: ***velocity vector***, $\mathbb{A}$: ***surface area vector***

This equation has many uses in the development of conservation principles, but in basic terms it is analogous to the filling of a bucket, where the quantity in the bucket is a function of what is inside the bucket to start plus the difference between the inflow and outflow.

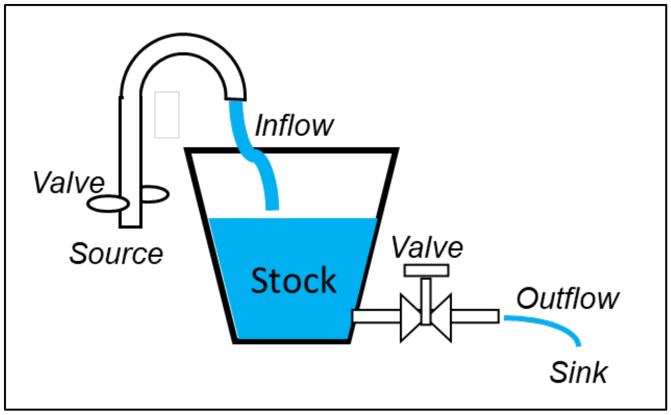


Figure 57: Control Volume Analogy

In SD terms this is simplified to the following mathematical relationships [51]:

Integral Equation:

$$Stock(t) = \int_{t_0}^{t} [Inflow(s) - Outflow(s)]ds + Stock\ (t_0) \tag{3}$$

Differential Equations:

$$\frac{d(Stock)}{dt} = Net\ Change\ in\ Stock = Inflow(t) - Outflow(t) \tag{4}$$

Mathematically, a complete SD model is a system of nonlinear differential equations. However, to reduce mathematical complexity and additional converters and connectors may be utilized to build and express relationships between elements as interlocking sets of simpler algebraic equations and ensuring dimensional consistency among the units of measure.[48, 61] By writing these equations in simple form in plain language, the causal structure posited is more clearly shown and provides greater overall model transparency. To develop these equations, modelers must use both theory, experimental data, and measured data (Figure 58). Therefore, SD requires both qualitative and quantitative inputs.

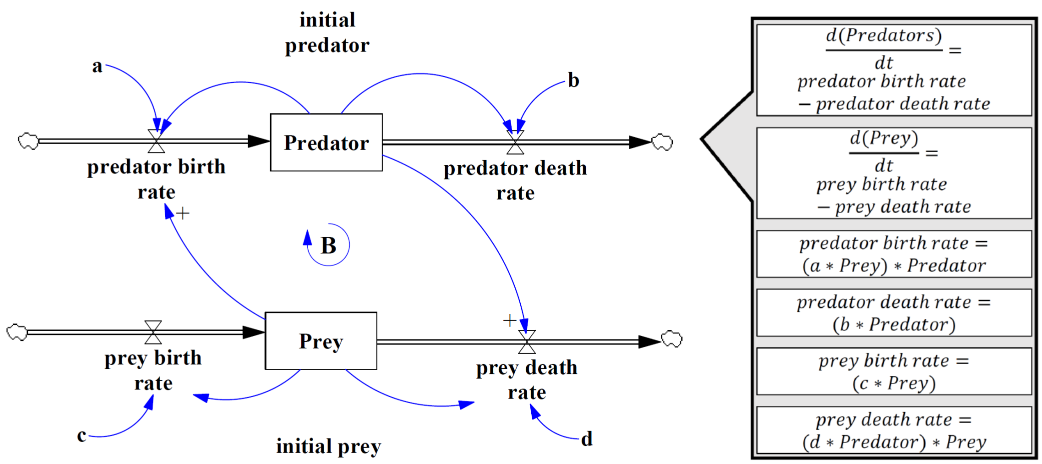


Figure 58: Classic SD -Prey (Lotka-Volterra) Example

SD models are likely to include more variables whose importance and inclusion are based on either logic or expert opinion, many of which may not have solid statistical estimates available [48]. It is therefore important to include the party 'customer' in the creation of the model from Step 1 to establish a level of confidence in the model to be able to accept the model as "an adequate depiction of the system it represents" [69]. Forrest posits that most SD models will only compete with mental models that decision-makers use to develop policy and make decisions in the absence of M&S. From his experience, he claims that customers will accept SD models as ‘valid’ due to the increased clarity an insight than what they previously had available to them.

The original SD software was DYNAMO, a programming language created by Alexander Pugh. Since that time, more advanced and user-friendly SD software has been developed such as STELLA (iThink) and Vensim which have contributed to the rise in interest in SD modeling [109, 113]. The later software makes the first three steps of the Forrester’s modeling process relatively easy by enabling computerized stock and flow diagrams, the ability to embed equations while building the model, and the ability to rapidly execute simulations with little computational power required—even large models can easily run on personal computers.

There are some important things to note about SD modeling that differentiates it from other modeling methods. Due to the mathematical relationships used and underlying integrals and differential equations, SD model variables are typically continuous and not discrete. The models work with aggregates and therefore the items in each stock are indistinguishable; they do not have individuality Though methods exist in modern software to treat some variables as discrete by implementing delayed pulse steps. As models of aggregate behavior, SD models will have broader model boundaries than other types. Therefore, modelers must think in terms of global structural dependencies and provide accurate quantitative data for those dependencies [90]. SD is likely “to include more variables based on the recognition by logic or expert opinion that they are important but for which solid statistical estimates may not be available” [48]. If data cannot be found or is unavailable “best guesses must be substituted until measurements are taken” [114].

In his 1991 chapter "System Dynamics and the Lessons of 35 Years" in the book *The Systematic Basis of Policy Making in the 1990s* [115], Forrester notes that "[e]ffectiveness of a model depends on how it uses the wide range of information arising from the system being represented." Forrester also notes that SD gathers its data differently than other social sciences in that it focuses on policies and structure as the basic building blocks for the model. As such, there are three primary sources of information used to build an SD model: mental data, written data, and numerical data. He argues that while data is often used to signify only quantifiable numerical information, the dictionary definition is broader in that it includes the phrase "material serving as a basis for discussion, inference, or determination of policy" [10, 115]. He posits that most human affairs are conducted mostly from a mental database then written knowledge and then numerical data.

The power of SD it the ability to generate, evaluate and assess trends; it cannot predict the future, nor can any model. Accurate trends are more important than exact data for past events—as no data can ever be found for future events. Alternative approaches involve using other M&S pedagogies to generate data that can then be aggregated and used in the SD model as mathematical data.

The SD paradigm is one of bounded rationality [116]. Bounded rationality for decision makers and modelers is defined by three overarching and unavoidable constraints. First, the information that is available regarding alternatives and consequences, is limited and often unreliable. Second, the human mind (and the model) is limited in capacity to evaluate and process the available information. Third, there is a limited amount of time to decide [20]. Ultimately this means that SD does not attempt to nor does it require all the variables in a complex problem. Rather, SD focuses on the variables that are essential to the problem and its context (the environment) defined by the modeler and the customer. Because it SD is a paradigm of bounded rationality, it does not attempt to seek a global optimum and instead must satisfice (accept an available option as satisfactory) by means of rules of thumb rather than explore every possible contingency [116].

3.3.4.2 Examples of Use

Since Forrester, first conceived SD, its use has expanded far beyond that for which it was originally envisioned. However, until Systems Engineering and Systems Thinking became popular coupled with user-friendly software developments in 1994, SD modeling was limited to a specialized group of experts [71]. Now, SD is used by Systems Engineers, the business community, in supply chain management, public policy, international and intra-national conflict, cellular receptor dynamics, [61, 116, 117]. Recently, the use of SD as a methodology for creating “simulation-based learning environments” has become more common and widely accepted amongst the SD community, grounded in the idea that the models can be used to improve decision-maker understanding of system structure and the essential features of system dynamics methodology and systems thinking.[118]

SD has seen some limited use in military applications, though typically only in academia. SD problems for the military include supply system analysis, workforce manning requirement assessments, operational impacts combat actions in insurgencies, combat models, project management, wargaming, and national stability to inform policy and analyze effects of operations [48, 73, 119-121]. Most notably, it has been used to study technology integration for Army PED by researchers at Charles River Analytics [122, 123]. However, SD does not play a significant role in the primary M&S used by operations and acquisition personnel nor for the creation of executable architecture.

CHAPTER 4. PROBLEM 1: OVERALL FRAMEWORK

From the stakeholder crosswalk and the literature review in 2.7, an overall framework was developed. This framework takes into consideration the systems concepts and important systems thinking considerations from Section 3.1 examined systems thinking, and IPPD (Figure 45).

Various modeling methodologies were reviewed, namely those of Arnold and Wade (Figure 38: Systems Thinking Systemigram from [67]), Sterman (Figure 48: Sterman's Iterative Modeling Process), Forrester (Figure 47: Forrester's 6-Step SD Modeling Process recreated from [69]) and Ford). In addition to verification/validation principles from Mavris (Figure 49 'Vee' Model recreated from [29]) With all of this in mind, the concepts are reflected in the framework presented in the Figure 59 on the following page.

Leveraging the AISR PED use case, the steps of this combined approach will be demonstrated. Figure 60 shows a mapping of the supporting research questions to the analytical framework. This mapping will serve as a guide for the remainder of the dissertation for the reader's reference and orientation. It is the authors position that the answering of these questions and demonstration of the framework via the use case is sufficient to provide an equivalency to the larger real-world problem. This method (and the M&S created) can be refined with data to provide a reasonable prediction of real-world operations and inform real-world decisions.

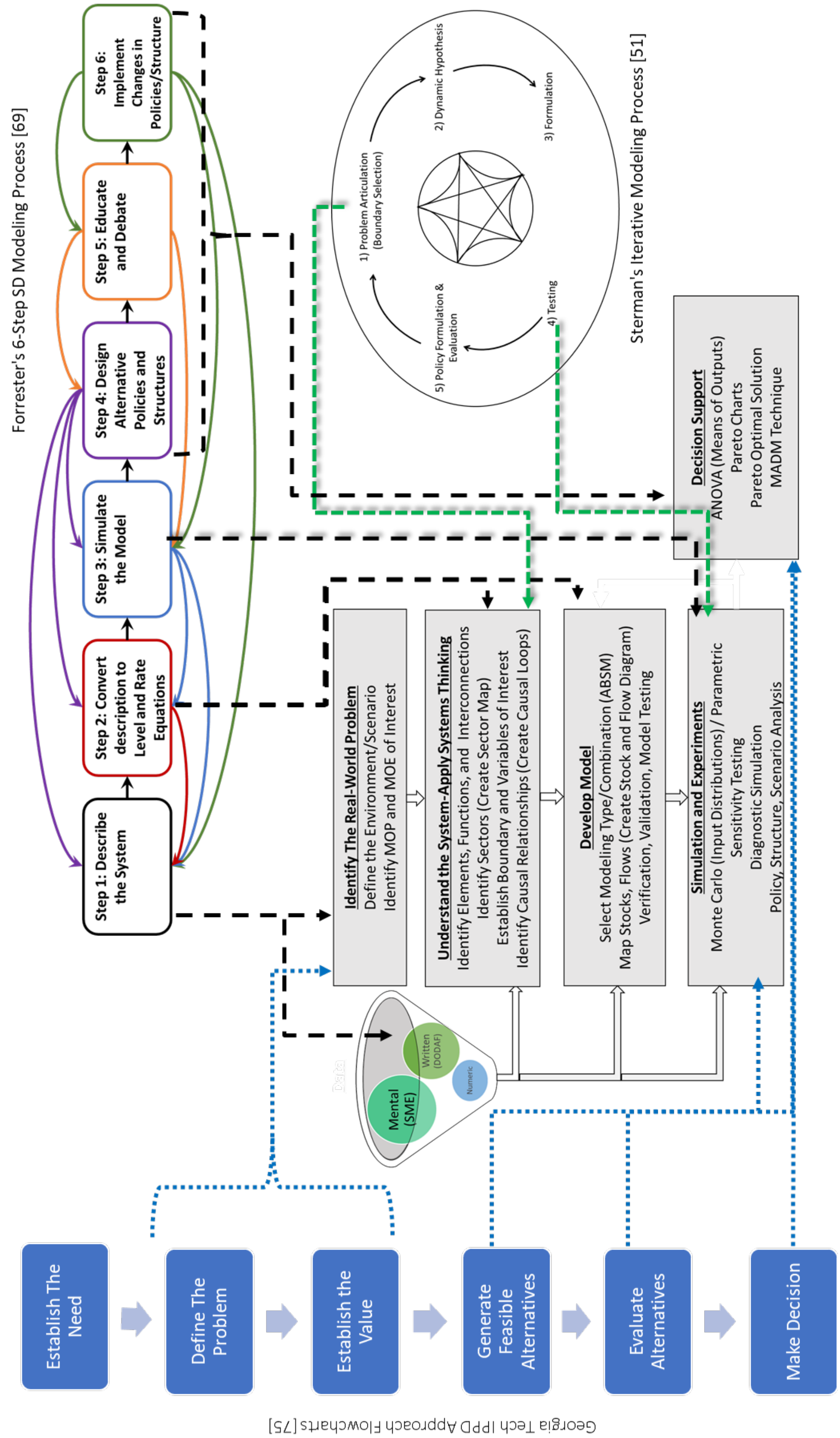


Figure 59: Overall Framework Diagram

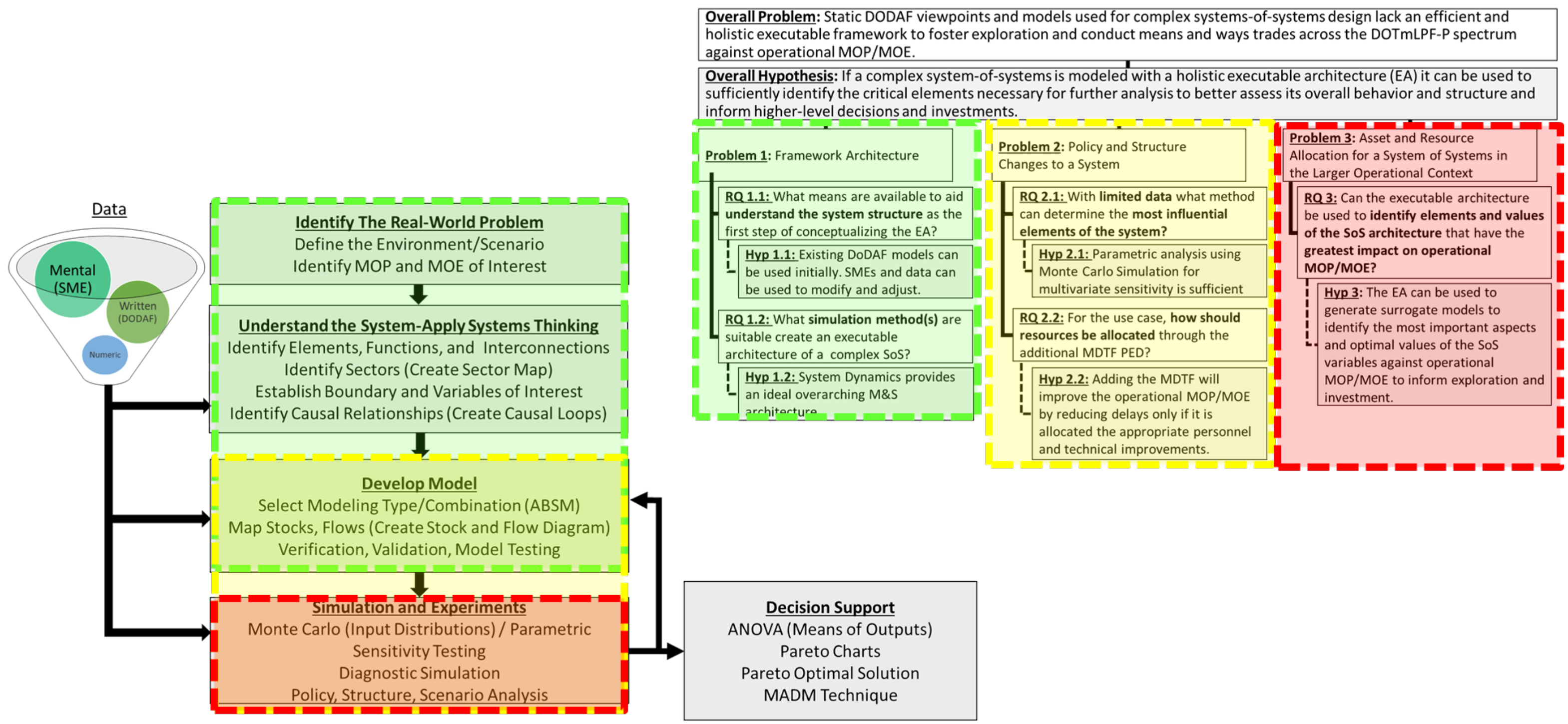


Figure 60: Mapping Research Questions to the Framework

4.1 Problem 1 Summary

Problem 1 as associated to the overall problem along with its subsequent research questions and hypothesis is depicted in the figure below for the reader's convenience. For detailed development of the sub-problem and questions see Section 2.5.

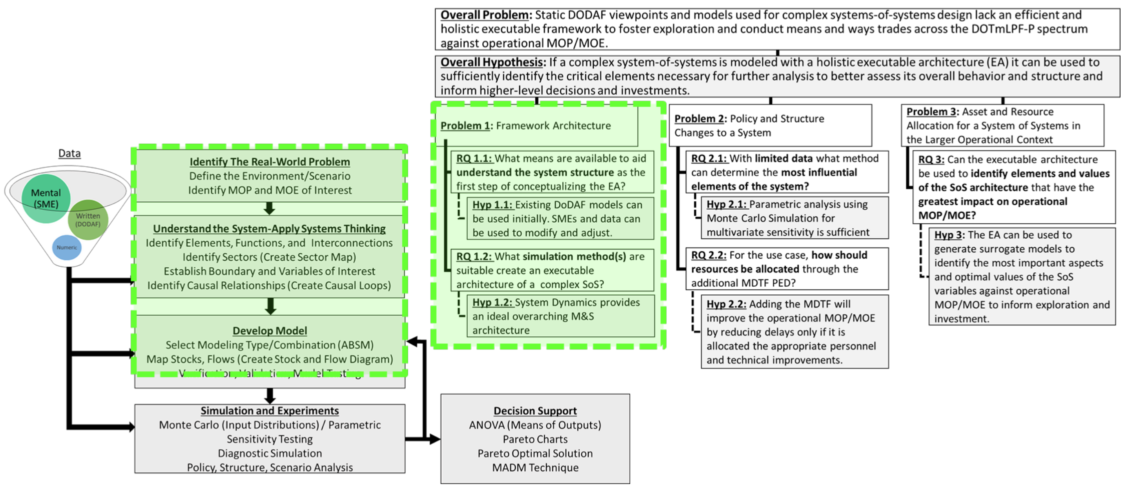


Figure 61: Problem 1 Summary

4.2 Identify the Real-World Problem/Scenario

As is true in every method, the first step is to identify and define the real-world problem. If the problem is static traditional modeling and numerical optimization methods may be the best course of action. However, if the problem is variant with time and exhibits dynamic behavior, then a simulation is required. For the purpose of this research a real-word problem was as a an need to dynamically model AISR-PED to allow the multiple stakeholders in the system-of-systems the ability to efficiently and holistically analyze the effects of modernization efforts and structural changes to the AISR enterprise architecture via a common operating picture with shared understanding to streamline communications and decisions.

4.2.1 Define the Environment/Scenario

For the given SoS, the modeler must define the environment and scenario in which the SoS will be operating. Defining the scenario is an important step to assess capabilities and test it against real objectives and forces (actual or simulated). Without an appropriate operating environment and scenario, one cannot assert the benefits or necessity of a capability [16]. Without defining the scenario and operating environment, one cannot identify the primary MOP and MOE. The method selected to generate the executable architecture should be flexible and adaptable enough to be able to alter the scenario and operating environment without drastic change or recreation of the EA framework.

For the purposes of this thesis, the use case for the executable architecture will focus on unmanned aerial intelligence and surveillance operations in support of long-range precision fires against near-pear threats in a multi-domain environment. Specifically, the future scenario focuses on Phase II, Seize the Initiative operations, (see Figure 177) during the first 48 hours of such a conflict during which forces must penetrate and disintegrate enemy A2AD systems and exploit the resulting freedom of maneuver to accomplish strategic goals (see Appendix A).

4.2.2 Aerial Intelligence Surveillance and Reconnaissance (AISR) Use Case

A system-of-systems is a set of interacting components organized to achieve a stated purpose that demonstrates operational and managerial independence of component systems as well as geographical distribution, emergent behavior, and evolutionary development processes [27, 64, 66]. Army Aerial Intelligence Surveillance and Reconnaissance (AISR) and the associated Processing, Exploitation, and Dissemination

(PED) of the information collected embodies this definition of a system-of-systems. The military phrase "sensor to shooter" implies a misleading simplification to the true complexity of AISR and PED which are both part of a complex system-of-systems spanning the globe. The massive addition of assets, sensor payloads, and architecture have been built up over the past decade as an attempt to not only provide more intelligence, but to make time sensitive intelligence accessible across organizations.

The complete system-of-systems operating in a continuous cycle (Figure 62). This system-of-systems includes forward-deployed systems and systems in sanctuary via reachback. Each of these systems and their associated personnel are independent organizations working in concert with one another under different operational and managerial controls, across vast geographical separation, and undergo independent system development. The complete system-of-systems, from a holistic sense, includes the aerial collection platforms with various collection of sensor payloads; the space assets that connect remote operation and transmit data back to sanctuary; the cyber distributed information systems network to provide access to raw data and intelligence products; the processing, exploitation, and dissemination systems and analysists that convert raw data to intelligence; and finally the "shooters" or kinetic assets that engage enemy forces.

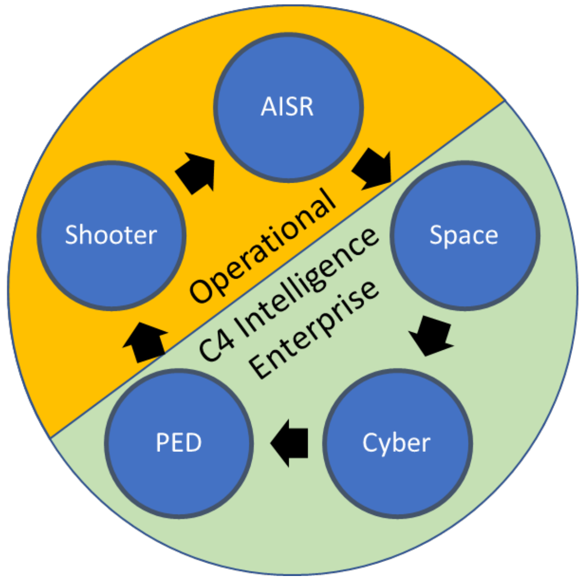


Figure 62: Aerial ISR System-of-Systems

After three years of exercises, war games, experiments, studies and field force observations the Army Campaign of Learning has identified the Army's number one capability gap (challenge) is the lack of adequate Intelligence, Surveillance, and Reconnaissance (ISR), specifically at echelons above brigade (EAB) in its ability to effectively support of large-scale combat operations (LSCO) against near-peer threats (NPT). The Army is addressing these challenges along three primary lines of effort from within the DOTmLPF-P spectrum: organization, materiel, and training as potential opportunities for improvement, thus making it a perfect candidate as a test case for this approach [124].

In typical modern wargames and simulations, AISR performance is simply included as a detection probability of enemy position to inform future wargaming or simulation agents' decisions. The AISR process itself, however, is typically not simulated in large scenario models nor is it wargamed. That is not to say AISR is not modeled at all. Many models and simulations exist to simulate UAS against advance enemy air defenses to determine detection probability, survivability, fleet mixing, and swarming tactics in an anti-access/area denial (A2AD) environment.[125] These modelers may focus on the number of assets, flight performance characteristics of different aircraft types, and sensor/payload capabilities in engagement or mission-level simulations to define future requirements or to analyze alternatives. These simulations typically focus on attrition rates of enemy and friendly forces and all rely upon the assumption that information is passed seamlessly between intelligence and kinetic assets, either assuming complete autonomy or simply neglecting the complex reach-back chain required.

Figure 63 from the Department of Defense (DoD) *Unmanned Systems Integrated Roadmap FY2013-2038* depicts the complex architecture associated with the collection of ISR information from a data transport perspective. From a broad overview, the AISR network consists of three layers: Space, Aerial, and Terrestrial Communications. The space layer, consisting of both protected military (MILSATCOM) and commercial (COMSATCOM) satellite communications relays, is critical for both the transport of raw data from manned and unmanned assets to the terrestrial satellite dishes that provide the gateway into the enterprise for further processing, exploiting, and disseminating (PED) of the data in the form of intelligence products. The MILSATCOM also provide critical beyond-line-of-site (BLOS) control of the UAS from either a forward deployed control shelter or via reach-back to a control shelter stateside.

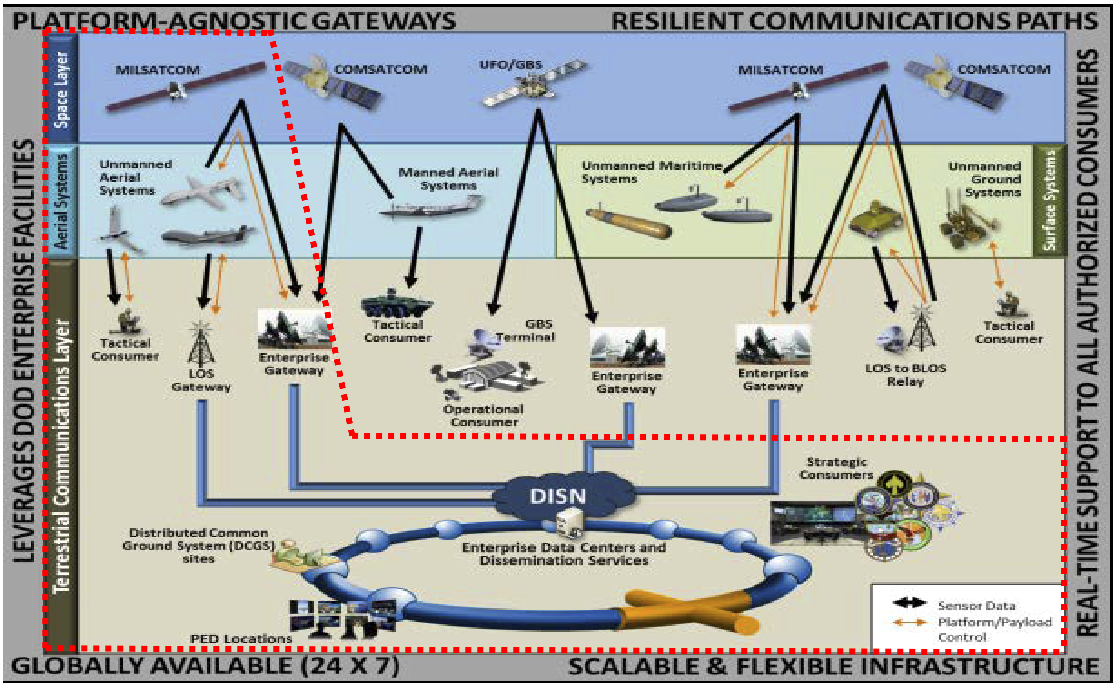


Figure 63: High-Level Command, Control, Communications, and Computers (C4) Operational Overview (OV-1) of ISR System-of-Systems [126]

While the fully autonomous, ‘drone,’ concept is imagined for future operations, it is an unlikely scenario in the near- to mid-term. Furthermore, it is unlikely that the military or the civilian leadership would ever accept kinetic actions without a human-in-the-loop element to confirm identifications and authorize weapons release [11, 127]. This means that this reach back architecture will remain relevant even if the UAS are fully autonomous. This complex loop requires extensive modeling and the application of systems thinking to analyze its complexity and improve its operation concurrently with the increasing and improving AISR assets.

To complicate matters, the Army’s current AISR reach back architecture was not a product of complete preplanned development through detailed creation and analysis of DoDAF products from the onset. Rather the fielding of the capabilities was evolved hastily to satisfy a Joint Urgent Operational Needs Statement (JUONS) during the War on Terror. New technologies and architecture were introduced and tested against real-world operations. To satisfy the ever-increasing demand for ISR, the DoD and the Army used a combination of existing programs of record, rapid modifications to existing platforms, and the rapid acquisition of commercial and contractor solutions. Unfortunately, while necessary, this enterprise grew in absence of unified plan under the DoD Architecture Framework. Aerial ISR and the PED architecture that supports it are constantly evolving.

As an experimental first step to make the ISR PED more agile for future near-peer conflicts, the Army has developed a conceptual unit deemed the Multi-Domain Task Force (MDTF) that is currently undergoing testing and evaluation in the exercises in the Pacific theater of operations, manual wargames, and simulations [42]. This pilot organization built around the 17th Field Artillery Brigade is equipped with high-mobility artillery rocket systems (HIMARS) and includes a plus sized battalion consisting of intelligence, information, cyber, electronic warfare and space elements (I2CEWS). The organization will have capabilities in each domain to help it penetrate enemies' anti-access aerial denial defenses and will be equipped with future technological capabilities, some of which are still in the conceptual phases [128, 129]. This proposed structural change has yet to be modeled into the existing framework analytically.

The current systems within the AISR-PED distributed architecture are each managed under different stakeholders, particularly Program Executive Offices (PEO), Program Managers (PM) within those PEOs and TRADOC Capability Managers, and three Futures Command CFTs each trying to develop, integrate and sustain their respective elements of the system. This distribution of responsibility requires an incredible amount of coordination and synchronization. Without an overall model of the governing system and idea of how it will be integrated and employed, modifications are developed by end user identified capability gaps developed through CBA and JCIDS. Improvements have been and will be based on observed or predicted gaps from SMEs against current or past operations without the ability to trace major influential factors, points of diminishing returns or vulnerabilities. This method results in "best guesses" for how to quantify questions such as how much technology may be needed to process data, how many people it can replace, how effective it is.

Systems thinking was likely applied by the Army throughout the AISR modernization process. However, given the fact that were so many entities involved over an extended period, and, it is reasonable to state that it was not examined holistically and simply grew too large with complex dependencies and non-linear relationships due to feedback loops cannot be readily visualized using traditional static MBSE. Before proceeding with additional modification and improvements, it is imperative the system be modeled holistically and examined to determine which factors are the most influential and a priority for future investment.

A system-of-systems this complex, however, is also difficult to model the entirety with popular methods used in military simulations such as discrete event simulations or agent-based models. Would an agent be needed for every aircraft, satellite, bit of data transmitted, intelligence analyst, kinetic fires system (ground and air), and every enemy on the battlefield? The computational time an effort would be unquantifiable and running thousands of cases with varying probabilities to account for stochasticity and uncertainty would be computationally expensive. Furthermore, validating and calibrating a model of this type so that end users would trust in the results is a challenge for a system in which data has been poorly recorded or is classified for a system that was developed in pieces rapidly over the past decade. A “soft” modeling approach may be more appropriate to provide qualitative understanding and support quantitative trend results to inform decision, policy, and investments.

4.3 Selecting Appropriate DoDAF for Executable Architecture

4.3.1 Research Question 1.1

The reader is reminded of the research question this section hopes to address:

Research Question 1.1

*What means are available to aid **understand the system structure** as the first step of conceptualizing the EA??*

Hypothesis 1.1: Existing DoDAF models can be used initially. SMEs and data can be used to modify and adjust.

Previous sections discussed the three primary sources of information recommended by Forrester [108] [115] for creating M&S for an executable architecture: the mental database, the written database, and the numeric database; with existing SME input, and DoDAF models to satisfy the first two requirements. He emphasized the mental database as the most important for determining know how and where information flows the roles of players in the system., and where decisions occur. The initial M&S can be generated and supplemented with numerical data as available.

Based on 17 years of experience in the field, the author suggests that AISR PED operations are no different, especially in the constantly evolving architecture compounded by the dynamic and uncertain operations of combat. However, this heavier reliance on mental and written data leads to the inclusion of many parameters which may be of assumed importance or have estimated or difficult to quantify values. Forrester [115] and Sterman [51] make an effort to note that excluding a variable from a model simply because accurate numerical data does not exist or cannot be found is more detrimental than estimating a value in a model for which behavior analysis is the primary goal. This is because removing the variable implicitly assumes that it has a value of zero and is not an influencer on the system, when through the application of systems thinking it was determined, at least initially, that the variable was important, or at least relevant.

4.3.2 Approach-Mapping the DoDAF

4.3.2.1 Initial Viewpoint/Model Selection

With these concepts in mind, the next step is to map static DoDAF viewpoint models to the dynamic simulation model. Unfortunately, DODAF 2.02 does not have product tailoring guides. Some DoD policy documents may specify specific DoDAF products for specific projects, which may at least provide a minimum set of diagrams, tables, etc. However, the decision of which DoDAFs to use is typically at the discretion of the architect based on what the team needs to convey the architecture [28].

DoDAF version 1.0 required a minimum core set of products required for every project:

- AV-1: Overview and Summary Information
- AV-2: Integrated Dictionary
- OV-1: High Level Operational Concept Graphic
- OV-5: Operational Activity Model
- OV-2: Operational Node Connectivity Description
- OV-3: Operational Informational Exchange Matrix
- SV-1: System Interface Description
- TV-1: Technical Standards Profile

While DODAF policy changed in later version, as not all the views were suitable for every problem, this list, while obsolete, at least creates a starting point on where to begin. Furthermore, DODAF 1.5 [88] provided a chart with applicable architecture products for various recommended uses and problem types. Bear in mind, however, that the some of the names and uses of these products have been modified or changed in DODAF 2.0 [89]. Nevertheless, Figure 64: DoDAF 1.5 Recommended Product Selection, provides a good starting point, despite its discontinued use. Even though the AISR TC-PED problem for study falls under C4ISR, we are assessing the system of system under an operational lens. Hence more operational views are recommended for operational planning and execution.

Uses of Architecture Data	All View		Operational View (OV)							Systems and Services View (SV)											Tech Stds View	
	1	2	1	2	3	4	5	6	7	1	2	3	4	5	6	7	8	9	10	11	1	2
Analysis & Assessment																						
Capabilities																						
- Gaps/Shortfalls					◉		●						●	●								
- Mission Effects & Outcomes, Operational Task Performance	●	●	●	●	●	◉		●		●	●				●	●			●	◉		
- Trade-Offs	●	●	●	●	●	●	●	●		●	●		●	●	●	●			●	◉	●	◉
- Functional Solutions	●	●	●	●	●	●	●	●		●	●		●	●	●	●			●	◉	●	◉
Operations																						
- Process Re-engineering	●	●		●	●		●	●														
- Personnel & Organizational Design	●	●	●	●	●	●	●	●	◉	◉	◉	◉			◉							
- Doctrine Development/Validation	●	●	●	●	●		●	●														
- Operational Planning (CONOPS and TTPs)	●	●	●	●	●	●	●	●		●	◉	◉	◉	◉							◉	
Systems/Services																						
- Communications	●	●								●	●	●							◉		●	◉
- Interoperability and Supportability	●	●	●	●	●	◉	●	●	◉	●	◉		●		●	◉	●			◉	●	◉
- Evolution/Dependencies	●	●									●	●	●		●		●			●	●	●
- Materiel Solutions Design & Development	●	●		●	●		●	●	◉	●	●	●	●	●	●	●	●	◉	◉	◉	●	◉
- Facilities Packaging	●	●		●			●	●		●	●		●	●							●	◉
- Performance							●	●						●		●			●			
Socialization/Awareness/Discovery																						
- Training	●	●	●	●	●	●	●	●		●	●	◉	◉	◉	◉							
- Leadership Development	●	●	●	●		●	◉	●		●			●	◉								
- Metadata (for federation)	●	◉																			◉	◉

● = Data Highly Applicable
◉ = Data is Often or Partially Applicable
(blank) = Data is Usually Not Applicable

Figure 64: DoDAF 1.5 Recommended Product Selection from [88]

Comparing a typical SD model to DoDAF, the holistic similarities between the operational viewpoints (OV) and SD modeling are readily apparent. Given that the overarching architecture for this research will be an SD model, a mapping of SD to DoDAF Operational Views is appropriate and feasible. In 2014, researchers at the Industrial University of Santander, Bueno et. al. [130] proposed an integrated executable architecture and mapped DoDAF operational views to SD stock and flow diagrams. The researchers proposed using only operational viewpoints to map static DoDAF products to a typical SD stock and flow diagram as depicted in Figure 65.

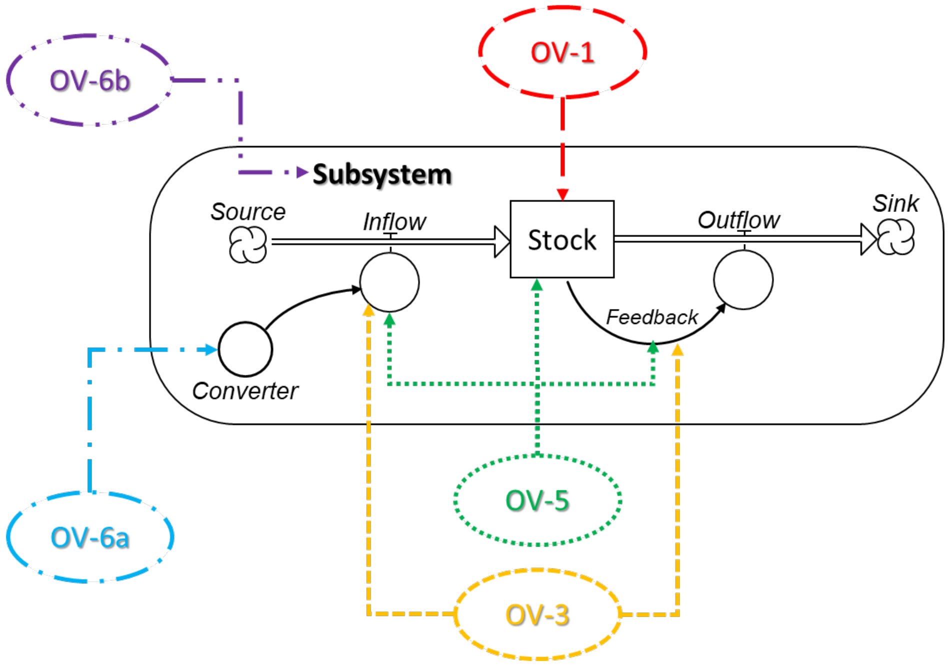


Figure 65: Mapping of OV Products to SD Model adapted from [130]

They suggested that a standard SD stock and flow diagram can be constructed from the information gleaned from just five different operational viewpoints: OV-1, OV-3, OV-5, OV-6a, and OV-6b. The ability to develop SD M&S using just these products is extremely valuable, given that operational viewpoints are some of the most often developed DoDAF products on Army projects, being produced for 86%, 71%, 86%, 57%, and 29% systems respectively [131].

However, in their conclusions, these computer scientists stated two findings. First, DoDAF models provide important and relevant information to the development of an interactive simulation. Second, DoDAF does not lend itself to automatically generated, direct mapping SD simulations, thus the modeler and SMEs must use them to glean relevant information [130]. This lack of automatic mapping, while an area of continued research, is not the emphasis of this thesis, quite the contrary. Recall the intent is to demonstrate the ability to rapidly and holistically develop a simulation that through its creation and through interactive experiments with the final product allow decisionmakers and SMEs to play games and produce results that inform decisions with some scientific backing and reasoning as opposed to a data-in-data-out black box.

4.3.2.2 High-Level Operational Graphic (OV-1)

In his doctoral thesis, *Architecture-Based Selection of Modeling Type For System-of-systems Analysis,* Dr. Burak Bagdatli [132] proposed a methodology for the SoS modeler to select a correct modeling technique by mapping various modeling paradigms to DoDAF via the viewpoint details and the modeling elements in a process he named Selection Of Logical Simulation Types for Systems-of-systems (SOLSTySS).

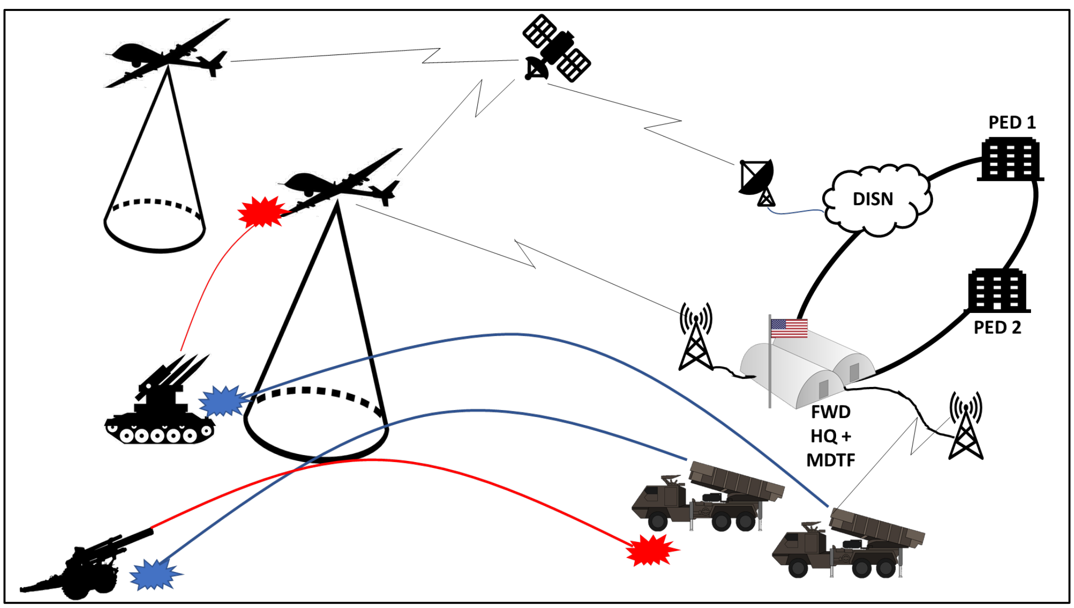


Figure 66: OV-1 for AISR PED in MDO With LRPF Example

In this work, Bagdatli found that the OV-1: High Level Operational Concept Graphic, is helpful in identifying stocks, flows and, potentially, variables while Bueno et. al found that OV-1 are useful for overall understanding of the system and for identification of stocks only. The author concurs with positions Bueno et al. and Bagdatli while also adding of posits that the OV-1 also provides information on the identification of the subsystems with the SoS framework, albeit at a superficial level. Using the OV-1 in Figure 66, the initial subsystems can be identified: enemy ground forces, friendly ground forces, friendly MDSS UAS, friendly PED (See Table 9).

Table 9: Initial Stock, Flow, and Variable/Converter Identification

Subsystem	Stocks	In-Flows	Out-Flows	Variables/Converters
Friendly UAS	MDSS UAS	Replacement UAS	Destroyed UAS	Crews
				Ground Control Stations
				Frequencies
				UAS Available
				Enemy AAA Fire
	Friendly Satellites	None	Destroyed Satellites	Enemy Satellite Attacks
				Satellites Available
Friendly Ground Forces	Friendly HIMARS	Replacement HIMARS	Destroyed HIMARS	Enemy Counterbattery Fire
				HIMARS Available
Friendly PED	Intelligence Backlog	Information In	Targetable Intelligence Out	MDSS UAS Overhead
				PED Personnel
				HIMARS Available
Enemy Ground Forces	Enemy AAA	Replacement AAA	Destroyed AAA	AAA Available
				Detection Rate
				Targeting Rate
				Friendly HIMARS
				Friendly UAS
	Enemy Artillery	Replacement Artillery	Destroyed Artillery	AAA Available
				Detection Rate
				Targeting Rate
				Friendly HIMARS
				Friendly UAS

4.3.2.3 Operational Resource Flow (OV-2)

Like Bueno, Bagdatli found limited use of the OV-2: Operational Resource Flow Description, due to lack of details in the OV-2 to inform the SD model. However, the DODAF website states that the:

> "OV-2 can be used to show flows of funding, personnel and materiel in addition to information. A specific application of the OV-2 is to describe a logical pattern of resource (information, funding, personnel, or materiel) flows."[31]

The OV-2 is specifically intended for supply chain analysis and the allocation of activities to resources, the former being a common usage of SD and the latter being one of the objectives of this research. The author posits that, the OV-2 aids in the identification of critical flows (personnel, materiel, etc.) that are necessary for the conduct of real operations. Too often such flows are neglected due to the difficulty of capturing such things in models and simulations. However, it is precisely these considerations that should be included to provide policy makers a better perspective of all that affects the operational effectiveness of the system; not simply the performance of the system in an isolated environment that treats these intangibles as unnecessary for the model or outside the scope of assessment. Of course, it depends on what problem the M&S is posing to address and what parts of the environment need to be included to answer that question while still reducing complexity. Hence, rather than dismiss the OV-2 outright, it should be considered to provide perspective to other stocks and flows in the problem that may need to be included as implied in Figure 64: DoDAF 1.5 Recommended Product Selection.

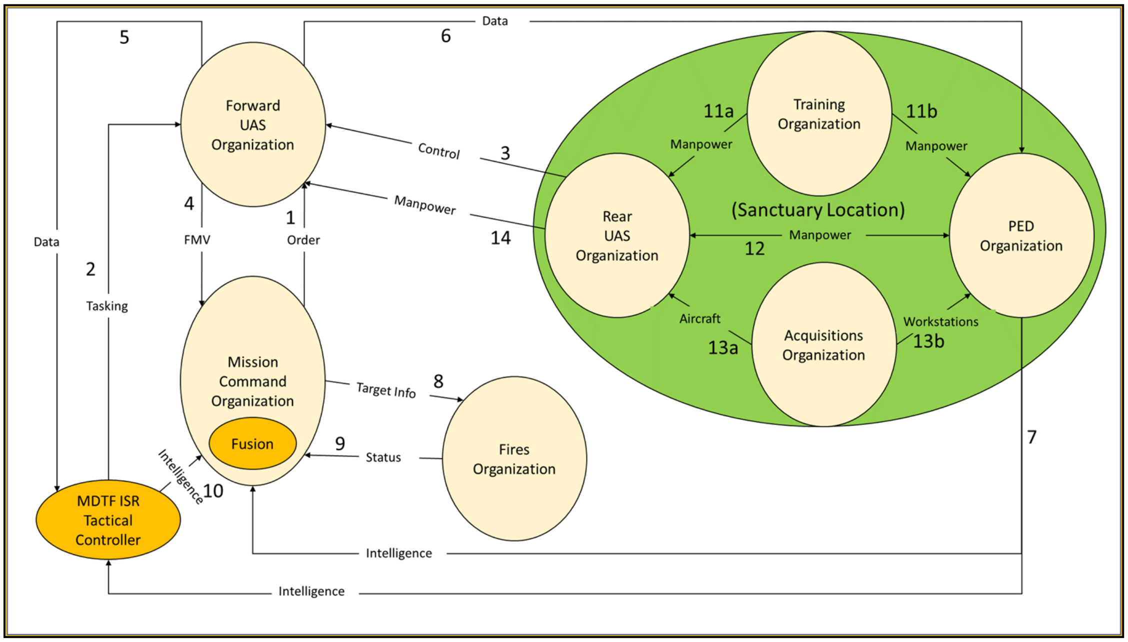


Figure 67: OV-2 (AISR-PED Example)

4.3.2.4 <u>Organizations, Activities and Resources (OV-3)</u>

Both Bueno and Bagdatli found that OV-3, which is inextricably linked to OV-2 could be used to inform flows and feedback, though Bagdatli notes that, like the OV-2, the implantation details in the OV-3 are lacking. It is true that neither given an indication of how much flow is required or the time rate of change of said flow but does aid in the identification of those flows that could affect stocks of interest.

Both sources and this author find that the OV-4, Organizational Relationships chart is of little to no value for the creation of an SD model.[130, 132] While organizational hierarchical relationships are important in understanding the roles in an operation, they provide no additional information to develop the SD model.

#	Operational Exchange Item	Sending Node	Receiving Node	Producing Operational Activity	Consuming Operational Activity
1	Mission Order	MC Node	FWD/Rear UAS Nodes	Create Collection Deck	Create Flight Schedule
2	In Flight Tasking	MDTF Node	FWD/Rear UAS Nodes	Send Updated Task	Provide ISR
3	Aircraft Control Data	Rear UAS Node	FWD UAS Node		
4	Raw FMV Feed	FWD UAS Node	MC Node	Send NRT Feed	Monitor Battlefield
5	Raw Data	FWD UAS Node	MDTF Node	Send Raw Data	Process into Intel
6	Raw Data	FWD UAS Node	PED Node	Send Raw Data	Process into Intel
7	Intel Products	PED Node	MDTF/MC Nodes	Create Intel Products	Fuse with other INTs
8	Target Information	MC Node	Fires Node	Identify Enemy Targets	Engage Enemy Targets
9	Status	Fires Node	MC Node	Engage Enemy Targets	BDA
10	Intel Products	MDTF Node	MC Node	Create Intel Products	Fuse with other INTS
11	Manpower	Training Node	Rear UAS/PED Nodes	Complete Training	Replace Shortfalls
12	Manpower	Rear UAS/PED Node	PED Node/Rear UAS	Cross train Intel PAX	Cross train Intel PAX
13	Aircraft/Workstations	Acquisitions Node	UAS/PED Nodes	Identify Shortfalls	Replace Shortfalls
14	Manpower	Rear UAS Node	FWD UAS Node	Identify Shortfalls	Replace Shortfalls

Figure 68: OV-3 Resource Flow Matrix (AISR-PED Example)

4.3.2.5 Operational Activity Decomposition Tree and Model (OV-5a, OV-5b)

The OV-5a is an Operational Activity Decomposition Tree and the OV-5b is an Operational Activity Model. Both Bueno and Bagdatli dismiss outright the usefulness of the OV-5a in the creation of an SD model but found the OV-5b as useful in identifying stocks, flows and feedback [130, 132]. However, according to the DoDAF 2.02 Architect Guide [89], these two documents combined are used to “describe the operations that are normally conducted in the course of achieving a mission or a business goal” and describe “operational activities (or tasks); input/output flows between activities, and to/from activities that are outside the scope of the Architectural Description” [31].

More importantly, the OV-5a and OV-5b are used to “uncover unnecessary operational activity redundancy” and “make decisions about streamlining, combining, or omitting activities.”[31] Part of creating an SD model is reducing complexity of the model, how this task will be accomplished with the M&S will be discussed in a subsequent section, however, the OV-5a which provides an “overall pictures of the activities involved” may prove useful as an additional manual check of redundancies as a “quick reference for navigating the OV-5b” [133].

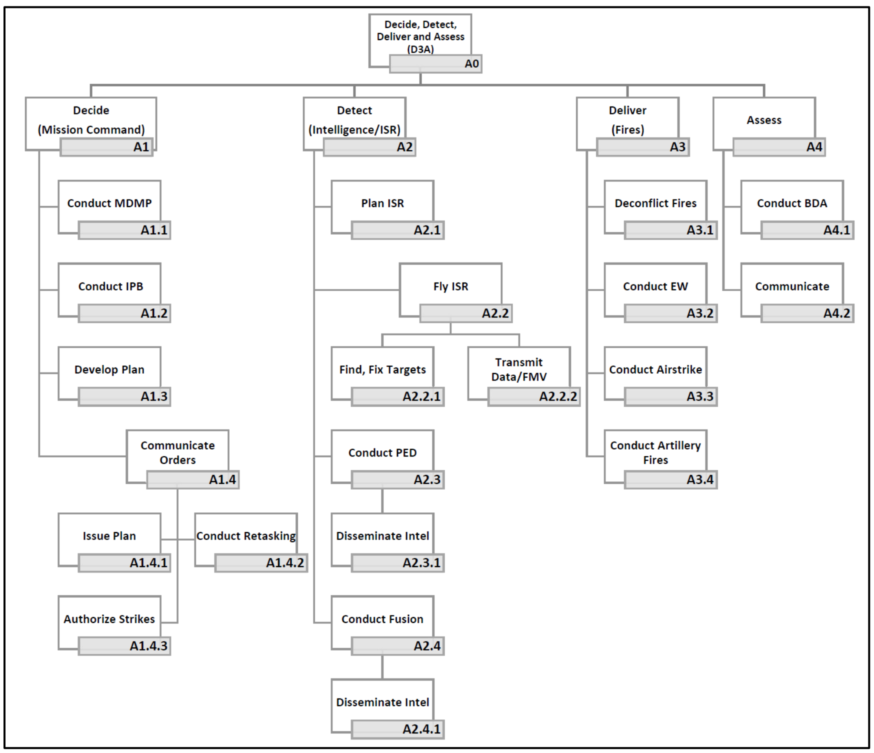


Figure 69: Example OV-5a Activity Decomposition Tree

The OV-5b depicts Activities connected by Resource Flows. As such, the OV-5b supports development of an OV-3. Together, the Resource Flows and Operational Activities contained in the OV-5b are linked to the Resource Flows identified in both the OV-2 and the OV-3 [133].

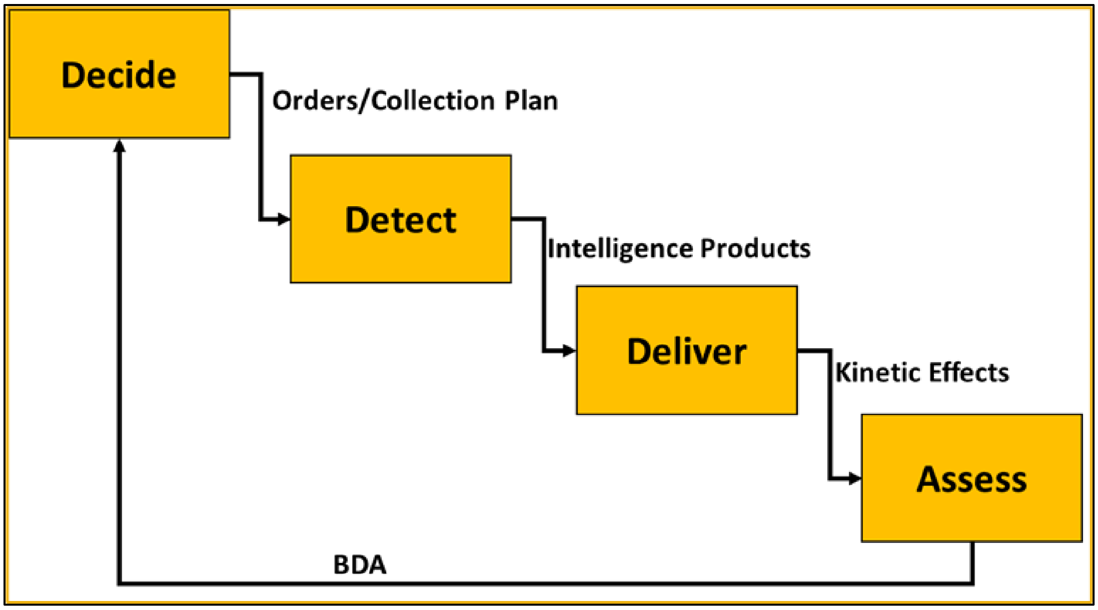


Figure 70: OV-5b Operational Activity Model (D3A Example)

4.3.2.6 Operational Activity Sequence and Timing Descriptions (OV-6a/b/c)

The OV-6, Operational Activity Sequence and Timing Descriptions are important to modeling dynamic behavior. Previous operational view artifacts were important to developing the model, but dynamic behavior over time (simulation) the modeler must have information as to the timing and sequencing of operational activities to inform parameters of the SD model. It is important to note that the OV-6a Operational Rules Model, OV-6b Operational State Transition Description, and OV-6c Operational Event-Trace Description are three different ways to represent timing and sequencing, hence, not all three need to be completed. Furthermore, these do not represent the only way to demonstrate sequencing and timing information[28]. The OV-6a "identifies and describes the rules and conditions that constrain activities within a described architecture [89]" that cannot be represented generally with a graphical MBSE depiction (see Figure 71). Bueno et al. [130] claim that OV-6a is important in identifying the governing parameters of the business model, which includes restrictions and rules on processes or activities.

A common use of SD is to evaluate the effect of system structure and policies on dynamic behavior over time. It would make sense, then, that an OV-6a be used to describe rules and policies in place that could be altered to change the dynamic behavior of the system to something more desirable. As such the author posits that the OV-6a is valuable for the creation of converters/variables and connectors and for the creation of the differential equations for the underlying relationships.

#	Applies To	Rule Specification	Rule Kind
1	MC Node MDTF Node FWD UAS Node Rear UAS Node PED Node MC Node Fires Node	Conduct 24-7 Operations for first 48-72 Hours	Activity
2	MC Node	Minimize loss of friendly assets UAS and HIMARS	Constraint
3	MC Node	Maximize destruction of enemy AAA and Counterbattery Assets	Constraint
4	MC Node Fires Node	Prioritize HIMARS fires against enemy targets	Constriant
5	UAS Forward Node	Launch replacements for downed UAS with minimal delay	Constraint
6	UAS Forward Node	Maintian selected operational readiness rate of UAS	Constraint
7	PED Node Fusion Node MDTF PED Node MC Node	Minimize intelligence overflow rates	Constraint
8	Cyber Node	Ensure satellite coverage over area of operations	Constraint
9	FWD UAS Node	Operations limited to combat radius of UAS	Constraint
10	UAS Forward Node MC Node	Conduct BDA in conjunction with reconiassance and targeting UAS operations	Constraint

Figure 71: OV-6a Operational Rules Model (AISR PED D3A Example)

Bagdatli notes that for a SoS, the OV-6b Operational State Transition Description is a good method to depict dynamics and maps well to SD since it shows how the system-of-systems works on a macro scale by use of a state transition diagram that graphically depicts how a node transitions its state as it responds to events.[132] Bueno et al. [130] note that the OV-6b is useful in identifying the key subsystems necessary to perform the essential function within which the stock and flow diagrams operate. However, Dam notes that the OV-6b does not provide clear linkage between function and data and cannot deal with complex interactions[28]. Dam also notes that all of the OV-6 diagrams "have a poor linkage between operational activities and information portrayed in the OV-5 [28]."

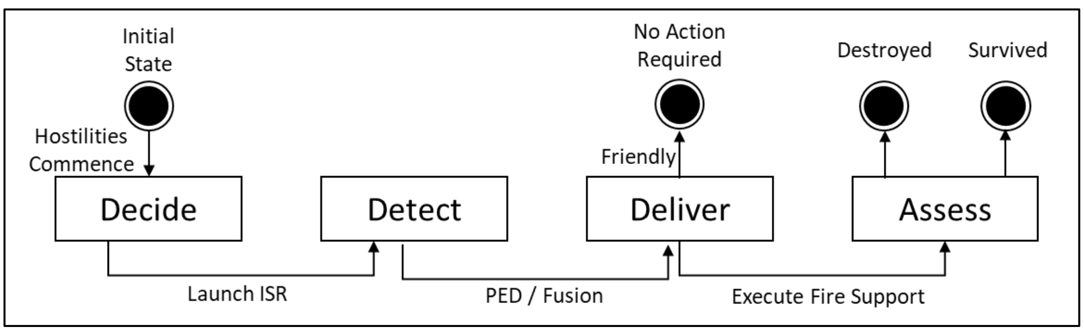


Figure 72: OV-6b State Transition Description (AISR PED D3A Example)

The OV-6c Operational Activity Sequences "identifies and describes a sequence of activities within a described an architecture [89]." Interestingly, neither Bagdatli nor Bueno et. al. recommends the use of the OV-6c. This view is used to capture time ordered information exchanges between operational nodes and provides a depiction of the sequencing of events in each scenario.

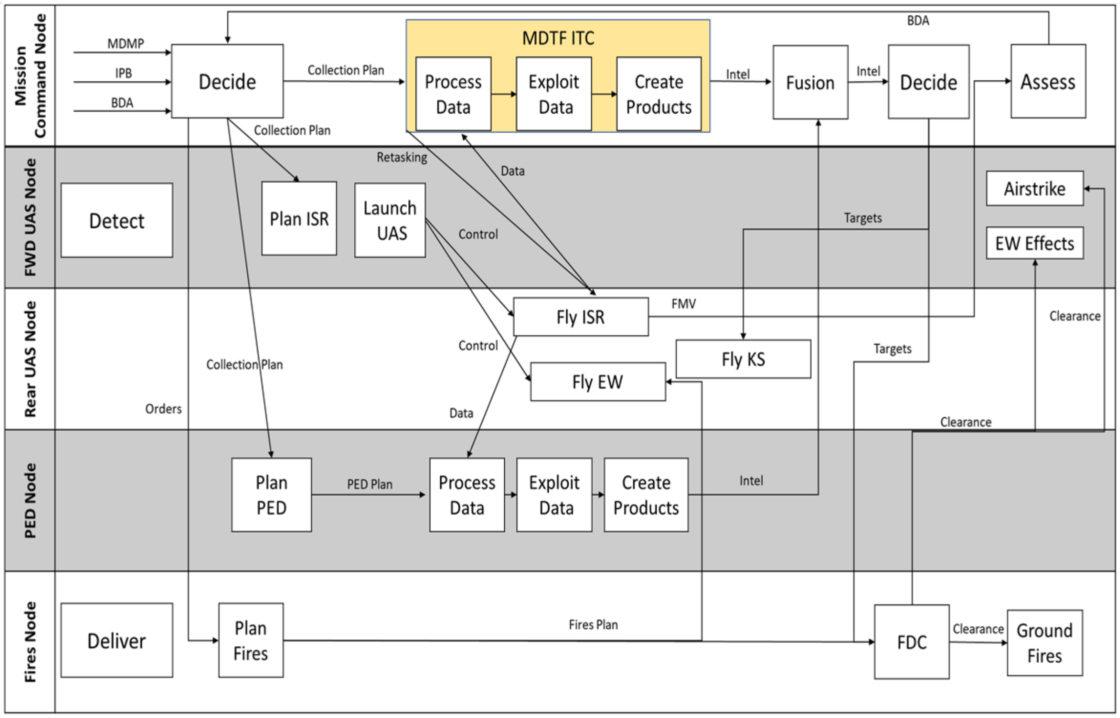


Figure 73: OV-6c: Operational Activities Sequence (AISR PED D3A Example)

4.3.2.7 System Interface Means (SV-2)

In addition to OV diagrams, Bagdatli recommends the use of System Views (SV) as well if available. In particular, he found SV-2 System Interface Means which is a representation of the primary physical connections between the systems of interest to be of particular value[133]. More specifically, the SV-2 depicts the “means by which resources flow between systems occur [89].” As such, the systems can be imagined as stocks and the connections as flow of the resources with ports as the variables that adjust flow rates with “perfect one-to one mapping.” [132] In addition to the mapping of the SV-2 to SD modeling provided by Bagdatli, the author finds that the SV-2 is also valuable in defining connectors that may not necessarily indicate a flow but as an information pathway that influences a flow particularly between subsystems.

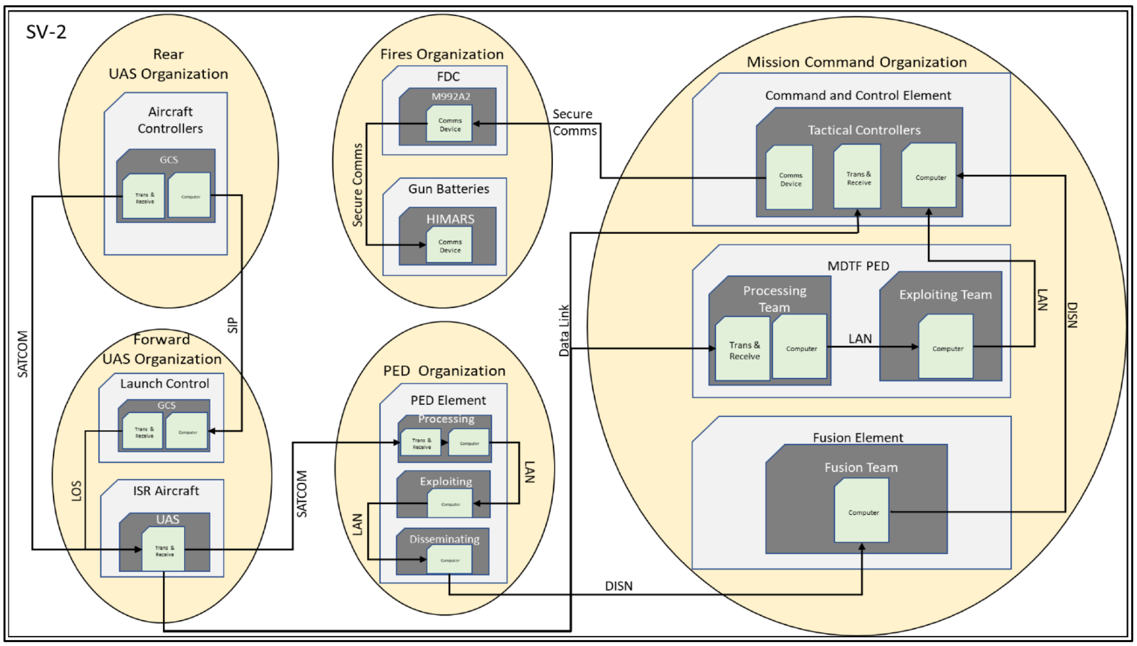


Figure 74: SV-2 System Interface Means (AISR-PED D3A Example)

4.3.2.8 <u>Systems and Services Resource Flow Matrix (SV-6)</u>

Bagdatli also notes the value of the SV-6, Systems and Services Resource Flow Matrix. This system viewpoint details the characteristics of the flow of resources between systems, the activities performed, and the rules and measures associated with these exchanges [31, 89]. While these attributes make for an ideal candidate for mapping to an SD M&S environment, the SV-6 really the “physical equivalent to the logical OV-3 table” [134] and provides very little useful information for the creation of an SD model over what is already contained in the OV-3 and the SV-2. Furthermore, the SV-6 is far less likely than the OV-3 to be generated in Army systems development (71% vs 29%). Therefore, the SV-6 can be used in the same manner as the OV-3 to identify stocks and flows but is redundant.

4.3.2.9 Systems Measures Matrix (SV-7)

The SV-7, Systems Measures Matrix (Figure 75) can be of great value in the creation of any M&S environment as it lists both the qualitative and quantitative metrics that are selected by the stakeholders/end-users and all of the measures that will be used in the model [31]. These measures contained in the SV-7 are those considered most crucial for mission success and are the performance parameters will must be met This document can include end-user defined MOEs and MOPs as measures that can be captured and presented in the SV-7 as performance parameters and requirements that can be developed and as specification that can be defined. This makes the SV-7 an important document in the CBA process during the creation of the ICD. However, it is not uncommon to continue to develop a complete set of performance parameters throughout the Architectural Description, as the performance parameters may not be fully known during early stages of architecture development. Hence the SV-7 is a 'living' document that "is updated throughout the specification, design, development, testing, and possibly even its deployment and operations lifecycle phases [89]." Surprisingly, this document is underutilized in the Army with surveys showing that it is developed for 0% of the systems in the study. In fact, it is the least developed product across the services, included in only 5% of systems developed [131].

System	Action	Metric	Value
UAS	Report Enemy Asset Locations (PIR)	Obserables/Hour/UAS	~1-5
PED	Process, Exploit, Disseminate	Reports/Hour	~20-30
Fusion	Generate Fuzed Targetable Intel	Targets/Hour	~10-15
HIMARS	Destroy Targets	Probability of Hit	50%
		Probability of Kill	50%

Figure 75: SV-7 System Measures Matrix (AISR-PED D3A Example)

In summary, through extensive literature review and study of the DOD Architecture Framework, the author recommends that if available, the OV 1/3/5b/6a/6b/6c in addition to OV-2, SV-2, SV-6 and SV-7 to build the preliminary model. After which, SME input and parametric values will be leveraged to refine the model based on the problem(s) to be addressed

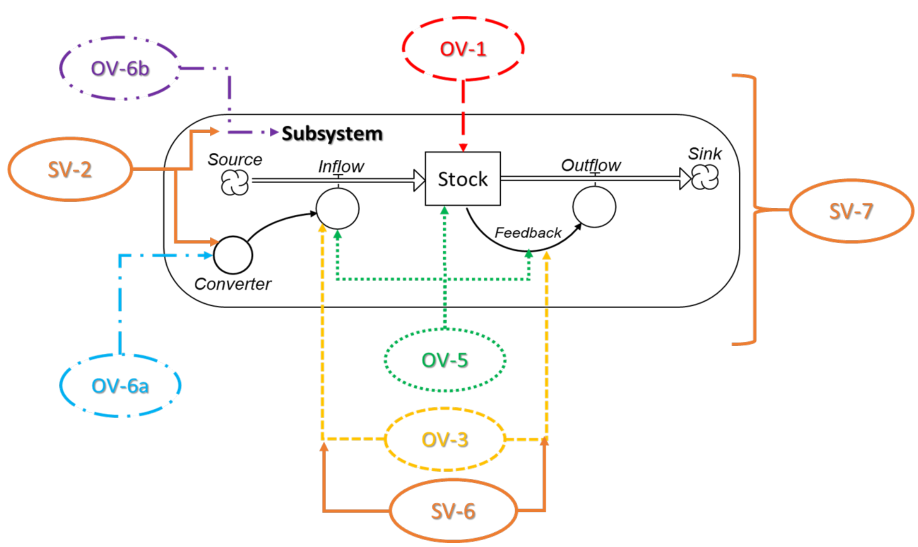


Figure 76: Final Proposed DoDAF Mapping

4.4 Understand the System/Apply Systems Thinking

As Forrester suggested, understanding the system requires data from three sources: mental, written, and numeric. As an initial starting point, SME input alongside available DoDAF artifacts/models are used to develop and understanding of the elements, functions, and interconnection within the system-of-system architecture. Numeric data will be used to modify and improve the model as available, otherwise, assumed values and parametric ranges will be used initially to ascertain behavior of the system under study.

4.4.1 Identify Key Sectors of the SoS

As an initial step, a general sector map is generated to depict the major sectors of the system-of-systems as shown in the Figure 77:

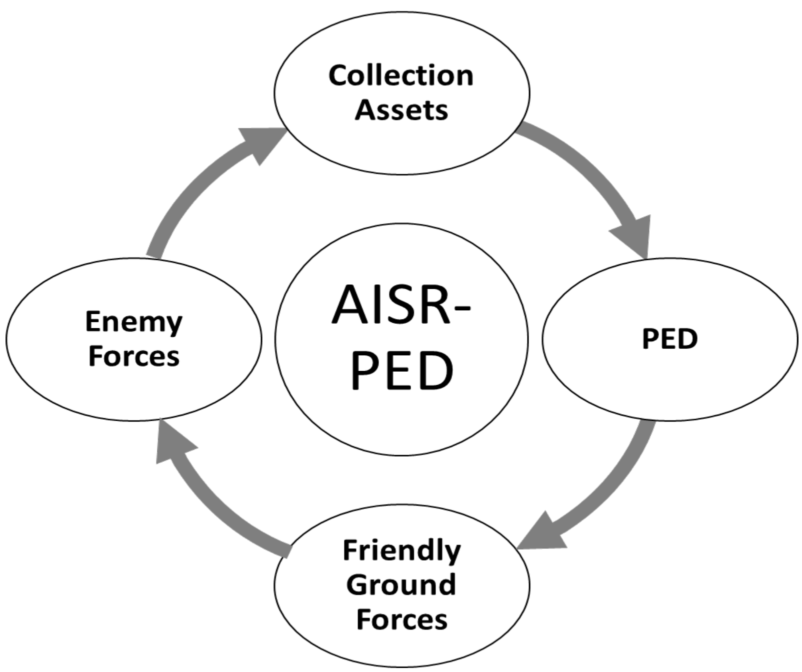


Figure 77: AISR TC-PED Sector Map

4.4.2 *Apply Systems Test*

The systems test shown in Figure 36: Arnold and Wade Systems Test is applied to gain understanding of the system or system-of-systems under study by forcing the modeler to identify elements, their functions, and the interconnections between them. For the given AISR PED SoS use case, is depicted in Table 10.

Table 10: Use Case Elements Functions and Interconnections

Layer	Element	Function/Purpose	Interconnections
Air Layer	Intelligence UAV	Find Enemy Targets	EWAC/KSAC/SAT/GCS/Enemy/Crews
	EW UAV	Jam Enemy ADA	KSAC/ISRAC/SAT/GCS/Enemy/Crews
	Strike UAV	Kill Enemy Targets	EWAC/ISRAC/SAT/GCS/Enemy/Crews
	Manned ISR	Find Enemy Targets	EWAC/ISRAC/KSAC/GCS/Enemy/Crews
Space Layer	MILSATCOM	Platform Control/ Data	UAVs/Gateways/Control Shelters
	COMSATCOM	Data Transport	Manned ISR/Gateways
Terrestrial Comms Layer	CGC	Fly UAS	UAS/PED/Command Cell/Crews
	Gateway	Relay Data	MILSATCOM/COMSATCOM/DISN
	DISN	Store/Disseminate Data	PED/DCGS/Gateway
	PED Centers	Process Data	DISN/Command Cell/DISN/ ISR Platforms
	DCGS	Disseminate Data	Command Cell/DISN/PED
Friendly	Command Cell	Decide	Fires/DCGS/ Aerial Assets
	Fires Element	Kill Enemy Targets	Command Cell/Enemy
Enemy	Enemy	Kill Friendlies	Aerial Assets/Fires
	Enemy ADA	Kill Friendly Aircraft	Aerial Assets/Fires

4.4.3 *Develop Causal Relationships*

From this architecture diagram and SME input, initial causal relationships can be developed. A sample of causal relationships are depicted in the figure below. Additional causal relations will be shown in the final model development in the next chapter.

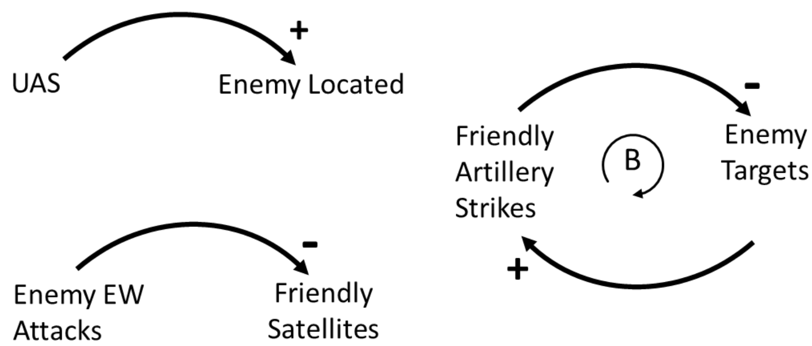


Figure 78: Sample Causal Relationships for Case Study

4.4.4 Establish Boundaries and Variables of Interest

The modeler must then, using the previously generated products, determine what variables are of interest and what factors will be included in the model using a model boundary chart like the one depicted in Figure 79 which has been generate for the use case in this dissertation. Recalling the definitions of endogenous, exogenous, and excluded from Table 5: System Modeling and Simulation Terms. The model boundary chart delineates the primary endogenous, exogenous, and excluded element and activities and elements in the system structure to be modeled.

This step is critical to identify model boundaries and aid in the identification of what does and does not need to be in the model. While this step does rely on SME and modeler subjective input, it is the first step in both utilizing systems thinking to reduce the complexity of the model and simulation. The endogenous elements that occur in the use case are identified in the innermost green circle.

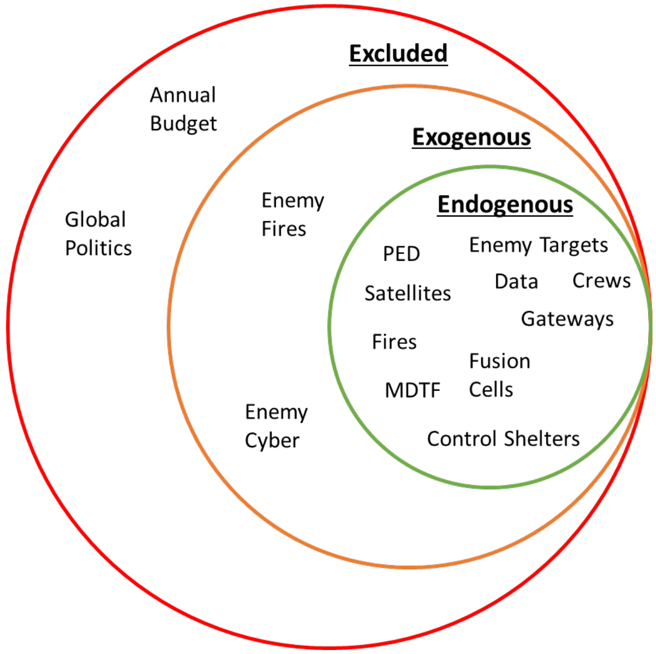


Figure 79: Model Boundary Chart

The orange middle circle encapsulates activities and elements that can affect the system of interest. Finally, the elements in red are excluded; while they may impact the system of interest indirectly, they are too far removed to be of interest for the study at hand. The identification of the key elements and activities will aid in the determination of the necessary simulation paradigm. Considering the variable of interest and their characteristics, as well as the model boundary and the role of the environment, the M&S type.

Using the Table 10: Use Case Elements Functions and Interconnections, the above sector map in Figure 77, and the section outlined in red in the OV-1 from Figure 175 an initial overview of the system-of-systems is generated and depicted in the figure on the following page:

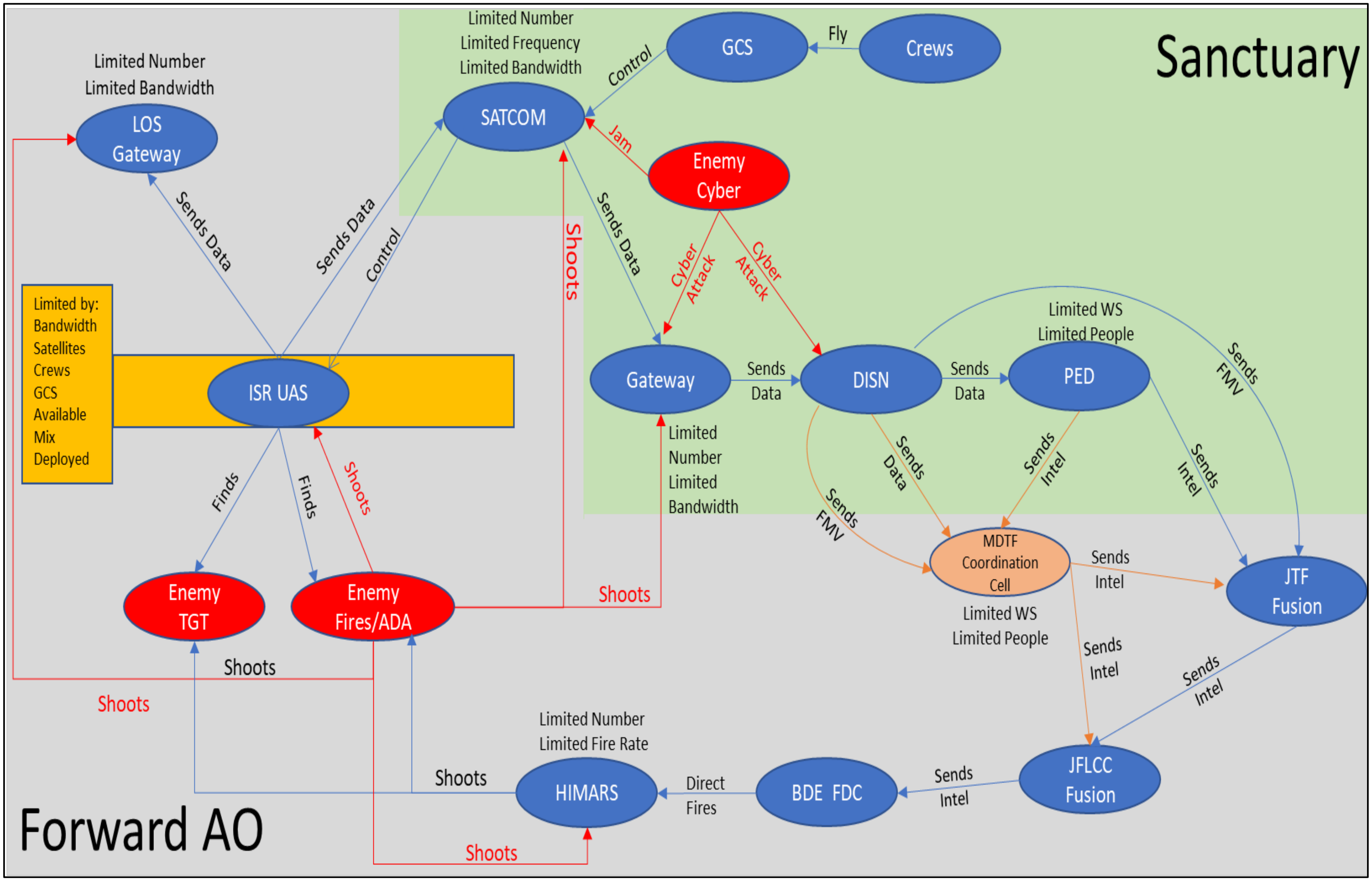


Figure 80: Case Study Architecture Elements and Interconnections Overview

4.4.5 Define MOP and MOE for the System

For the SoS under study, the key operational MOP and MOE of interest to the operational customer must be defined to ensure that the right M&S is built and that the M&S is right; else the executable architecture will be a waste of effort, fail to gain acceptance of stakeholders, and ultimately be excluded. Determining the MOP and MOE are critical for simulation paradigm selection; it is of no value to select a method that is incapable generating the desired observations and outputs.

Up until this point, the author has used the terms MOP and MOE in the technical sense of the terms. However, there is ambiguity in the definitions of these terms between the technical/acquisitions branch of the military, and the operational branch of the military. It is important to both define MOP and MOE as well as differentiate technical and operational MOP/MOE for clarity for the remainder of this dissertation.

The collaborate publication "Technical Measurement (INCOSE-TP-2003-020-01)" [21] coauthored by members of the U.S. Army, PSM (Practical Software and Systems Measurement), INCOSE (International Council of Systems Engineering), and Lockheed Martin, defines technical MOP as :

> **Definition: <u>Technical Measure of Performance:</u>** "The measures that characterize physical or functional attributes relating to the system operation, measured or estimated under specified testing and/or operational environment conditions [21]."

When dealing with the development of technical systems these technical MOP translate to system performance targets or goals, e.g. data transmission rate, max range, endurance, etc. They compare how well, from a supplier's viewpoint, the delivered system performs against system level requirements. But while targetable performance metrics are sufficient for development, the crux of the issue and the ultimate question is how these measurables translate to the desired real-world mission success. This concept leads to the MOE which [21] defines as:

Definition: **<u>Technical Measure of Effectiveness:</u>** "The "operational" measures of success that are closely related to the achievement of the mission or operational objective being evaluated, in the intended operational environment under a specified set of conditions; i.e., how well the solution achieves the intended purpose.[21]."

Operational MOE from a technical definition perspective can have a wide range depending on the problem definition. These MOE are stated from the customer's standpoint and represent the most important indicators of affordability, suitability, and performance needed to achieve mission success across the life cycle of the system. These MOE are typically normalized to aid comparison of alternatives and evaluate achievement of key operational performance, e.g. maintenance-free operating period, systems/dollar/year [21]. These "operational" measures of success are typically predicated off metric that presume an operational advantage or benefit in a conflict or scenario.

However, the operation branch of the military defines MOP and MOE differently. The Joint Publication 3-0, *Join*t *Operations* defines the MOP as:

Definition: <u>Operational Measure of Performance:</u> "A criterion used to assess friendly actions that is tied to measuring task accomplishment [135]."

In an operational environment these operational MOP amount to measurable activities such as number of hours flown, number of reports processed, number of sorties flown, etc. These operational MOP are easy to track and brief to higher commands as indicators of success.

However, while these metrics indicate how well a unit or a system performs, they give no indication of the value the asset/system/unit provides relative to the success of the commander's ultimate objectives. In other words, MOP may sound good, but if they had no bearing on the desired end state that it is nothing more than a number. For example, if 1000 intelligence reports were processed, but none contain any information of value that led to the destruction of enemy targets, then the intelligence collection was not effective.

Definition: <u>Operational Measure of Effectiveness:</u> "A criterion used to assess changes in system behavior, capability, or operational environment that is tied to measuring the attainment of an end state, achievement of an objective, or creation of an effect [135]."

When deriving MOP and MOE perspective is vital (supplier versus customer; technical versus operational). From the preceding definitions and discussion, operational MOP can be conflated with technical MOP with the operational MOE being estimated or assumed. It is important that distinctions are made. When assessing a SoS architecture and changes therein, operational stakeholders (customers) need to be able evaluate a SoS for how changed across the DOTmLPF-P spectrum affect not only operational MOP, but more importantly operational MOE.

Like the environment and scenario selection, modelers should seek guiding documents such as doctrine, Army Vision, Army Priorities, scenario objectives, and subjective SME guidance. For the AISR PED use case, the intention of ISR and improvement to the SoS is to increase the collection of intelligence that leads to the elimination of enemy targets while sustaining friendly assets while reducing chronic fatigue on limited capabilities (intelligence personnel, aircrews, HIMARS, and UAS) and intelligence losses due to expired information (LTIOV). From the resources and discussions with AISR PED SMEs following MOP and MOE are identified:

Table 11: AISR-PED Measures of Performance

Friendly UAS Losses Over Time
Friendly HIMARS Losses Over Time
Friendly Satellite Losses Over Time
Enemy Anti-Aircraft Artillery (AAA) Losses Over Time
Enemy Artillery Losses Over Time
Intelligence Processing Rates
Intelligence Overflow Over Time

Table 12: AISR PED Measures of Effectiveness

Enemy to Friendly Artillery Loss Ratio Over Time
Enemy AAA to Friendly UAS Ratio Over Time
Total Loss Ratio Over Time
Friendly Loss Percentages Over Time
Intelligence Overflow Ratios
Average Intelligence Capacity Utilization
Intelligence Personnel Overwork

4.5 Problem 1.2 Executable Modeling and Simulation Approaches

The reader is reminded of the research question this section hopes to address:

Research Question 1.2

Which simulation methods are suitable to create an executable architecture of a complex system-of-systems?

Hypothesis 1.2: System Dynamics can provide an overarching M&S architecture that can capture key aspects of the system and enable understanding of technology benefits and limitations, policy impacts, and the likely outcome of future investment strategies.

As discussed previously, the selection of an appropriate paradigm should focus on seeking the correct level of aggregation, though the end result may be a blend of paradigms. Initially, however, the modeler must select an overarching simulation paradigm to create the broader architecture at the strategic level. Recall that the intent of this framework is to identify general trends that can be further informed by smaller, higher fidelity models only for elements of the system that are the most influential as a means to reduce complexity and increase efficiency of M&S creation and experimentation so as it is not excluded from the CBA process or the developing Futures Command methodology.

This subsection addresses Research Question 1.2 shown in the Figure 61. this specific section seeks to demonstrate a method to compare and evaluate different major modeling and simulation paradigms in greater detail. First each paradigm will be described in more detail, with a description, general principles, and examples of use. The different paradigms will then be compared against criteria developed from both Army modeling and simulation guidance and developed from observations regarding desirable traits of executable architectures and stakeholder requirements.

4.5.1 Determine Necessity for Simulation

Prior to selection of a simulation paradigm, one must first determine if simulation is even necessary, recalling the challenges of DoDAF and the limitations of static MBSE architecture, especially when dealing with complex systems-of-systems. Table 7 discussed Dr. Jerry Banks' circumstances for when simulation is or is not an appropriate tool [78]. As a matter of example, the AISR-PED use case will be used against these criteria. This will not only support the need for simulation, but the answers aid in the selection of modeling paradigm.

In Table 13, the premise of an executable architecture for AISR-PED is evaluated against circumstance for which simulation is an appropriate tool. For circumstances that apply, the associated row in the center column is annotated with the word 'yes' and the cell is color-coded green; for those in conflict, the associated row is annotated ‘no’ and color-coded red. Because we are looking at circumstance which are appropriate for simulation, green and yes are desirable. Additional remarks are included with corresponding hyperlinks to areas in this document that expound upon the concepts further. It can be seen from this comparison, that the problem overwhelmingly supports simulation as opposed to analytical solutions or static models. The cells with ‘no’ do not preclude the use of simulation but should be briefly addressed.

First, when constructing an executable architecture, it is unlikely that an analytical solution exists to be verified. This does not preclude the model from being simulated but can pose challenges in validation of the model. Additionally, while the EA is intended to provide some of the same benefits of wargaming as a learning tool, its primary intent is to analyze the SoS not simulate warfighting.

The premise of an executable architecture for AISR-PED is evaluated against circumstance for which simulation is not an appropriate tool. For circumstances that comply, the associated row in the center column is annotated with the word ‘yes’ and the cell is color-coded green; for those in conflict, the associated row is annotated ‘no’ and color-coded red. Because we are looking at circumstance which are not appropriate for simulation, red and ‘no ‘are desirable. As previous, remarks are included to provide insight as to the applicability.

Of note, are three circumstances which have the potential to indicate that simulation is not appropriate; these ideas should be addressed and help formulate additional criteria for the selection of a simulation paradigm, or combination thereof. First, the cost of the simulation must not exceed the potential savings. As noted previously, the simulation for the EA should not be computationally or monetarily expensive, lest it is likely to put additional strain on the budget and the timeline. This premise directly applies to the concept of time and personnel to verify and validate the model.

Table 13: Evaluating if Simulation is Appropriate for Use Case

Simulation is Appropriate	Y/N	Remarks
To study and experiment with complex system internal interactions	**Yes**	AISR-PED is highly complex. Internal interactions between systems must be analyzed.
To simulate the effects of organizational changes on a model's behavior.	**Yes**	Determining effects of organizational changes must be evaluated against operational MOP/MOE
When knowledge gained from the design of the simulation would identify and suggest ways to improve the system	**Yes**	One of the goals for the EA is to improve understanding of the system-of-systems through the development of the simulation.
When changing inputs and observing outputs can provide insight into the most important variables interactions.	**Yes**	The EA should be interactive to allow univariate and multivariate changes to observe operational MOP/MOE impacts
When it can be used as an educational device.	**Yes**	EA should facilitate tabletop discussions and similar benefits to wargaming
To prepare for future events by experimenting with new designs or policies prior to implementation.	**Yes**	The EA should be able to evaluate changes in policy and structure to be able to test them prior to costly implementation or live experimentation.
To verify analytical solutions.	**No**	It is unlikely that analytical solutions exist for complex SoS in an operational environment.
To develop requirements by simulating different capabilities for consideration.	**Yes**	The granularity of the EA will determine the fidelity necessary for the capability being evaluated.
As a training device with reduced cost.	**No**	EA is not intended to be used as a training device.
For visualization of an operation	**Yes**	The EA should provide visualization of system interactions and effects on operational MOP/MOE
When a modern system or SoS is so complex that interactions can only be evaluated through simulation.	**Yes**	Complex SoS may have second and third order reactions and feedbacks that cannot easily be evaluated without simulation.

If the effort is too great for the EA, it is likely to be excluded Lastly, poorly recorded or classified data may add difficulty in creating and validating the M&S. This framework demonstrates that, like Forrester noted, there are three types of data: mental, written, and numeric; with existing SME input, and DoDAF models to satisfy the first two requirements. The M&S can be supplemented with numerical data as available.

Table 14: Evaluating if Simulation is Not Appropriate for Use Case

Simulation is Appropriate	Applies		Remarks
If the problem can be solved using common sense.	**No**		Complex SoS interaction and feedback loops are difficult to infer. Static DoDAF models are useful for detail tracking but not for visualizing effects
If the problem can be solved analytically	**No**		No equations can be used to solve complex dynamic stochastic architecture,
If it is easier to perform direct experiments.	**No**		Direct experiments of complex systems are time-consuming and expensive and require huge monetary expense up front
If cost of simulation exceeds potential savings.	**Yes**	**No**	Dependent on type of simulation used. Goal is to create an inexpensive simulation relative to the cost of acquisitions.
If no data or estimates are available to inform the model and simulation	**Yes**	**No**	Data or estimates can be challenging to obtain especially for systems that have yet to be created/implemented or are classified. However, simulation should be capable of providing insights with the three primary data sources and refined with additional data.
Not enough time or personnel to verify and validate the model.	**Yes**	**No**	Some simulations can be extremely difficult to verify or validate. Simulation should provide reasonable behavior, not predict exact behavior for non-physics-based models.
If managers expectations are too high or if abilities of simulation are overestimated.	**No**		*EA serves to provide technical orientation* and *management orientation;* it does not predict the future.
If system behavior is too complex it can't be simplified, i.e. human behavior.	**No**		For a strategic level EA average values or perceived trends can be used. For areas most influential, additional simulation types may be used to provide greater fidelity.

With the recommendation for simulation for the problem set being confirmed, subsequent sections will provide an overview of the four major modeling paradigms researched for this study.

4.5.2 *Select Modelling and Simulation Type/Combination*

After careful contemplation of the problem, examination of available data, and an initial attempt to understand the system and reduce complexity using systems thinking, a M&S type or combination thereof must be selected. An M&S paradigm must be selected to fit the problem, along with the appropriate scenario to assess the capabilities and behaviors under examination.

Table 15: Questions to Evaluate Analytical Approaches from [37]

Question/Criteria
1. Can the approach evaluate the doctrinal approaches you have collected?
2. Can the approach estimate the measures of effectiveness you are using?
3. Can the approach represent the scenarios, tasks, and functions identified in the study definition?
4. Does the approach represent the correct warfighting scope?
5. How large a team does the analytical approach require to execute?
6. How much analytical overhead (i.e., estimation of outcomes not relevant to the CBA) must be absorbed in the approach?
7. How long will the approach take to execute?
8. Does the approach require construction of a set of special-purpose models? If so, how long will it take, and will it be difficult to win acceptance of these models?
9. Is the approach agile enough? Can it quickly assess many alternatives (US and enemy CONOPS, scenarios, and capabilities)?
10.What is the backup plan if the approach does not work?

As implied in Figure 46: M&S Characterizations, models and their variables can be dynamic or static, deterministic or stochastic, discrete or continuous, and the M&S method should be capable of simulating said characteristics. When considering large scale systems-of-systems for strategic-level insight, the question of necessary aggregation is an important one. Additionally, the DoD *Capabilities-Based Assessment Users Guide* lists a specific set of questions to evaluate analytical approaches that must be considered summarized in Table 16. Each of these questions should be considered when developing the model and simulation. Determining the proper complexity of a model is often a challenge. If a model is too detailed (high-fidelity) it becomes too large and complex to effectively be used as a communication tool to promote shared understanding; too simplified (low fidelity) and it ceases to be able to answer detailed questions.

The quote commonly attributed to Albert Einstein, "Everything should be made as simple as possible, but not simpler" demonstrates the ideals standard for models and simulations. While there are many modeling and simulation types, this thesis will examine four types in detail: Petri Nets (PN), Discrete Event Simulations (DES), Agent Based Modeling, and System Dynamics. Figure 81 provides an overview of these modeling types and their typical levels of abstraction, variable types, and general characteristics.

Developers of M&S seek foresight and insight, seeking qualitative predictions and quantitative predictions of the problems they seek to investigate.[103] This pursuit requires the appropriate level of aggregation and modeling paradigm for the problem being investigated, each with their own unique assumptions, methodology, and different criteria to evaluate the results. All too often, practitioners of M&S resort to the M&S paradigm with which they are most familiar.[93] But the type of problem will dictate the level of aggregation required based on the desired insights and the characteristics of the problem being modeled, which requires the correct M&S paradigm. Dr. Sterman makes it a point to stress to the model builder that the creating a model of a system is not the singular purpose;

solving a problem is and the model must be simplified to the appropriate level of complexity to accurately replicate a specific problem.

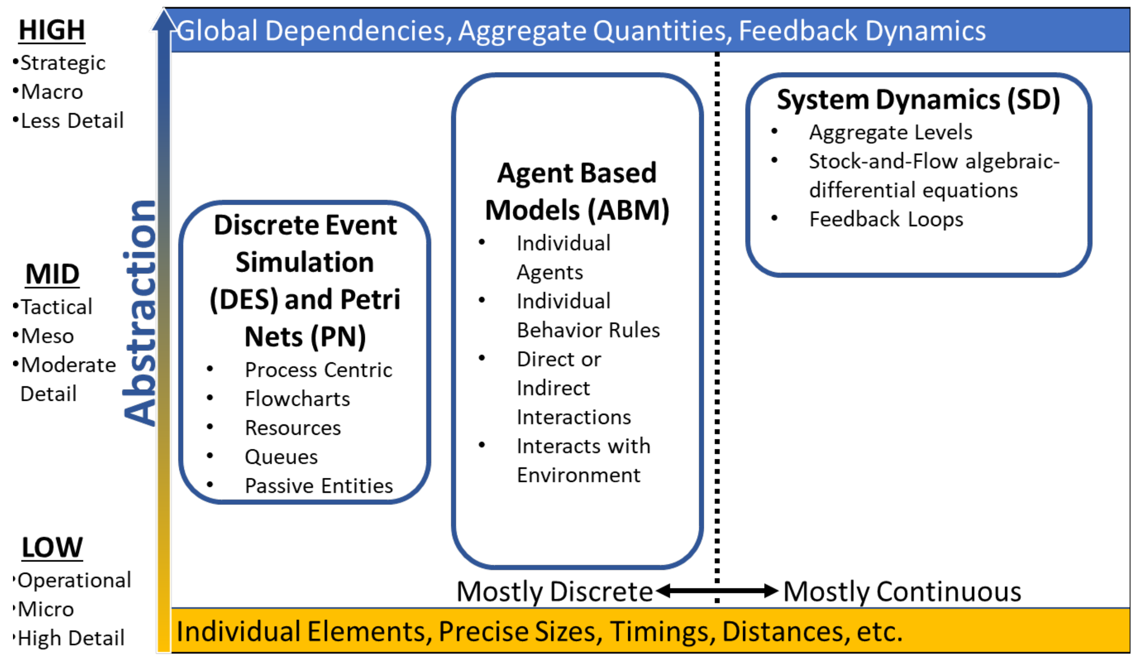
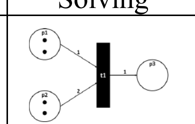
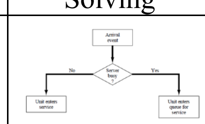
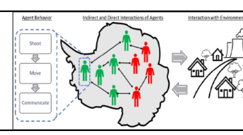
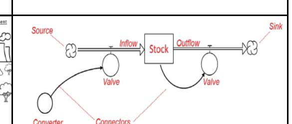


Figure 81: M&S Paradigms adapted from [90]

A modeler may but does not necessarily have to choose between "modeling the forest or modeling the trees," i.e. macro or micro. While such a distinction is possible and sometimes necessary, one may model both the model and the trees by combining modeling methods for various aspects of the problem depending on the aggregation required or in the absence of data to provide emergent behavior of micro elements to inform the macro model. Rather than make the distinction of petri nets, discrete-event simulation, agent-based modeling, or system dynamics the correct answer may be a combination of two or more of the methods.

For quick reference, a brief summarized comparative chart of the paradigms explored in the previous section is depicted below in Table 16: M&S Paradigm .

Table 16: M&S Paradigm Comparative Summary

	Petri Nets	DES	ABM	SD
Approach	Process view	Process view	Bottom-up	Top-Down
Detail Level	Meso	Meso	Micro-Macro	Macro
Level	Individual (hetero)	Individual (hetero)	Individual (hetero)	Aggregates (homo)
Time	Discrete	Discrete	Discrete	Continuous
Stochastic or Deterministic	Stochastic	Stochastic	Stochastic	Deterministic
Mathematics	Graph Theory	Event Bootstrapping	Logic Rules	Integral Equations
Model Elements	Queues & Activities	Queues & Activities	Agents and Logic	Equations, stocks, flows, feedback
Relationships	Linear	Linear	Non-Linear	Non-Linear
Source of Dynamic Behavior	Events	Events and delays	Events	Stock levels, delays and feedback
Goal	Explore impact of randomness and complexity	Explore impact of randomness and complexity	Explore emergent behavior	Explore global structural dependencies.
Application	Problem Solving	Problem Solving	Exploring	Problem Solving
Graphic				
Hyperlink	Page 168	Page 105	Page 110	Page 115

As noted by Dr. John Sterman Dr. Hazhir Rahmandad, forcing oneself to simply choose between the paradigm can be “a false choice rooted in confusion.” The answer is the seek the right level of aggregation [136]. However, the modeler will have to compare and contrast the benefits and limitations of the aforementioned paradigms against the objectives of the study, the available data, and the time constraints, amongst any other criteria the modeler deems important for the given problem.

Despite the generalized usages implied in Figure 81 hybrid approaches are not limited to using system dynamics as the macro and agent-based as the micro. Multiple studies have demonstrated that some paradigms can replicate the performance of others or can be imbedded in each other, some using system dynamics within an agent-based model or discrete event simulation and some with agent-based models or discrete event simulations within the larger system dynamics model [93, 100, 104, 137, 138].

For the case study, the selection of a paradigm or blend of paradigms will be selected in the subsequent chapter as well, leveraging the systems thinking tools already developed and displayed in this chapter. The subsequent chapter will also detail the methods to construct the model using available data and validation and varication (V&V).

4.5.2.1 Initial Comparison

Beyond just characteristics of the variables, goals, and treatment of time for simulation paradigms shown in Table 16, the reader is reminded of the ten questions posed by the DOD *Capabilities-Based Assessment Users Guide* depicted in Table 15: Questions to Evaluate Analytical Approaches from [37]. As an initial attempt at determining the appropriate M&S for the executable architecture, the paradigm are assessed against an abbreviated form of those questions in Table 17.

Table 17: Initial Assessment of M&S Types for Study

Question/Criteria	**CPN**	**DES**	**ABM**	**SD**
Ability to evaluate the doctrinal approaches	Y	Y	Y	Y
Ability to estimate the desired measures of effectiveness	N	N	N	Y
Able to represent the scenarios, tasks, and functions identified in the study definition	N	N	N	Y
Able to represent the correct warfighting scope	N	Y	Y	Y
Size of team required to execute analytical approach	M	L	L	S
Time to execute approach	M	L	L	S
Requires construction of special-purpose models?	Y	Y	Y	Y
If so, how long will it take	M	M	M	S
Will it be difficult to win acceptance of the model?	U	U	U	U
Agility; able to assess many alternatives	N	N	N	Y

Each of the methods can evaluate the doctrinal approaches, though it depends on the scale and the metrics being observed. For the AISR PED use case, CPN and DES are good for tracking the flow of intelligence along with sources of delays but are not good at depicting feedback from second and third order effects. ABM can assess engagements and individual actors to determine battle damage for actual assets and kinetic engagements but for the scope of the problem would require far too many agents to be created. Even if aggregated, things like data flow would be difficult to model and assess. SD on the other hand, is capable of modeling both the asset interactions, engagements, and data flow, but again, at a higher aggregation. Techniques could be used to discretize elements that need more specific granular data points like UAS and HIMARS assets.

The ability to estimate MOE would depend on exactly what the desired MOE are. If the MOE are based solely on improved data flow or intelligence reports DES and CPN would be suitable, if not ideal. If kinetic effects are the key MOE, ABM would provide engagement-level interactions to inform global effects. However, if the impact of intelligence and the effects on friendly forces and enemy forces are desired, SD would provide a holistic sinew to join the two. Data flow would be aggregate, which is acceptable for strategic-level observations. However, it cannot simulate battle engagements beyond statistical elements or stochastic effects. Those areas could be refined by integrating an ABM. This assessment is also true for represent the scenarios, tasks, and functions identified in the study definition. Given the MOP and MOE for the AISR PED use case and the description of SD characteristics, SD appears to be the only paradigm that is able to capable of satisfy in the requirements.

The size of team required to execute analytical approach is also a functin of the fidelity required and the scope of the operation under study. For example, for a campaign-level or mission analysis (Figure 8: DOD Modeling Hierarchy from [35]) with high levels of aggregation, small to medium team would be necessary, whereas if the scope was large and the fidelity was high with many entities, attributes, and activities, the effort could be massive. For a typical SD problem, small teams are recommended to meet with SMEs, review policies, and develop architectural frameworks because SD focuses on global relationships at the strategic level of macro aggregation. If areas needed additional blended modeling types, additional members may be required.

Because the CBA ICD development timeline is short, the intent is to develop an EA using a rapidly developed and executed method to obtain reasonable behavior models of the system. CPN, DES, ABM can all be developed in a moderately reasonable amount of time with the right expertise and available simulation platforms such as AFSIM. However, SD is marketed as being able to be rapidly developed and has low computational requirements, meaning many alternatives can be assessed quickly.

Agent-based models are the easiest to understand because they deal with the actions and rules of individual agents. However, agents are not suitable for all aspects of the given AISR PED use case, namely data transport. CPN and DES can be confusing and serve, essentially, as black boxes to decision-makers. Because SD is a graphical interface that is easy to view and understand and utilizes easy to understand algebraic and differential equation written in plain English with instantaneous visualization of outputs and effects, the author believes it will be easier to gain acceptance of the model. This belief is strengthened by the fact that it emphasizes behavioral trends and influence on those trends rather than numerical results.

From this initial selection method, the author deduces that SD would provide the best method to create a strategic-level holistic executable architecture of the SoS. However, to provide numerical objectivity to the selection of the M&S paradigm, a classic Pugh matric with absolute weighting is used.

4.5.2.2 Pugh Matrix

A type of matrix diagram, a Pugh Matrix (PM) is an easy to use decision support tool for multi-attribute decision making. This method is intended to help reduce subjectivity when comparing several alternatives. The method allows for the pair-wise comparison of the alternatives against a baseline rather than against each other for many decision criteria to help reduce subjectivity [139].

The first, and arguably most important step of crafting the PM is the selection of evaluation criteria. An incorrect, inadequate, or incomplete selection criteria can reduce the quality of the assessment and lead to an incorrect decision [139]. For the selection of an appropriate M&S paradigm for the creation of an EA capable of informing strategic-level acquisition, policy, and allocation decisions four primary selection criteria were identified: ability to provide technical orientation, ability to provide managerial orientation, flexibility/adaptability, and cost. These four primary selection criteria were further divided into18 secondary selection criteria depicted in Figure 82. These criteria were generated using the simulation criteria found in Table 13 and Table 14; from desirable EA characteristic generated from gap analysis and through discussions with SMEs.

The second step was to select a baseline to compare the other paradigms against. As discussed in Section 3.3.1.3 CPN and TPN have been used in the past to create executable architecture by Levis, et. Al. [30, 87] and in Executable Architecture Methodology for Analysis (EAMA) by Pawloski et al [140]. Additionally, PN in general were originally mentioned specifically for the creation and testing of OV-6 series DoDAF models. For these reasons, CPN were selected as the baseline paradigm.

For the third step, each alternative paradigm was then compared against the baseline criterion by criterion. For each criterion, the baseline was assigned a numeric value of ‘3’. A pair-wise score with ‘3’ means the alternative is the same as the baseline, ‘4’ means it is better, and ‘2’ means it is worse. To add extra levels of discrimination, ‘5’ means the alternative is much better and ‘1’ means it is much worse.

Criteria		Pugh Concept Selection Matrix	Weight	Colored Petri Nets	Discrete Event Systems	Agent Based Modeling	System Dynamics
				Paradigms			
Criteria	Techinical Orientation	Identify Emergent Behavior	3	3	2	5	4
		Capable of handling large number of alternatives	4	3	2	1	5
		Ease of use	4	3	3	4	5
		Identify Delays Due to System Structure	3	3	3	2	3
		Knowledge Gained from Design of Simulation	4	3	3	2	5
		Develop Requirements	4	3	2	2	4
		Repeatable, Traceable	5	3	3	3	3
	Managerial Orientation	Evaluate Policy	4	3	3	3	5
		Incorporate Operational MOP/MOE	5	3	3	5	4
		Play Games/Interactive Experimentation	4	3	3	3	5
		Strategic Level	3	3	3	4	5
		Easily communicable	5	3	2	4	4
	Flexible/Adaptable	Easily adjusted for structural changes	4	3	4	2	4
		Data dependency to construct	3	3	3	5	4
		Incorporate Intangible difficult to quantify variables	2	3	3	3	4
	Cost	Computationally Inexpensive	4	3	2	1	3
		Monetarily Inexpensive	3	3	3	2	4
		Rapidly Developed	5	3	3	3	4
		TOTAL SCORE		54	50	54	75
		WEIGHTED SCORE		207	191	207	287

Figure 82: Pugh Selection of M&S Paradigm

The numeric values are tallied at the bottom. From the initial computation, SD is the clear winner based on the criteria, just as the hasty subjective analysis in the previous sub-section. CPN and ABM had equal scores and slightly edged out DES. This is an interesting result given that the two M&S paradigms are used to simulate very different things. However, this result provides great insight into the best methods to use to refine certain aspects of the global EA for areas deemed the most influential and for which smaller models can be used to confirm global assumptions in the absence of data. If desired, hybrids could be created, and the Pugh Matrix could be reevaluated; this is a form of qualitative optimization [139]. Because the intent of this study is to demonstrate a framework for creating an overarching, rapid, EA to evaluate operational MOP/MOE at the strategic-decisionmaker level, the use of hybrid models is beyond the scope of this thesis but will be explored in future work.

To improve differentiation, the criteria can be weighted. While there are several types of scaling approaches (absolute, sales-driven, analytic hierarchy process, etc.) the absolute scale from 1-5 was used for this study. Each numeric value represents the relative importance of the criterion to the stakeholders. The value of with '1' represents the criteria being of minor importance, '2' being of moderate importance '3' being important to the stakeholder, '4' being very important, and '5, being of extreme importance. The weights were individually assessed for each criterion and applied. The weight value is multiplied by the pair-wise comparison value for each criteria and alternative paradigm before being summed at the bottom to calculate the weighted score.

Coincidently, not only did the weighted score ranking of the alternatives match that of the non-weighted score, but CPN and ABM again have matching scores. Furthermore, the ratio of the scores is identical as well. Ironically, this is not a miscalculation nor was it done intentionally: simply changing one of the weights changes the values and ratios. For example, if the first criterion "Identify Emergent Behavior" was changed from a weight of '3' to '4' the scores change to 210, 193, 212, and 291, respectively.

4.5.3 Selection of System Dynamics

For purposes of this study, SD was selected as the appropriate type of M&S to analyze and conduct experiments on the AISR PED architecture problem. Given the previously generated products from the use of Systems Thinking, stock and flow diagrams can be created and then combined with causal loops to create the SD model. A sample of stock and flows for this problem for the primary variables of interest are show in the figure below.

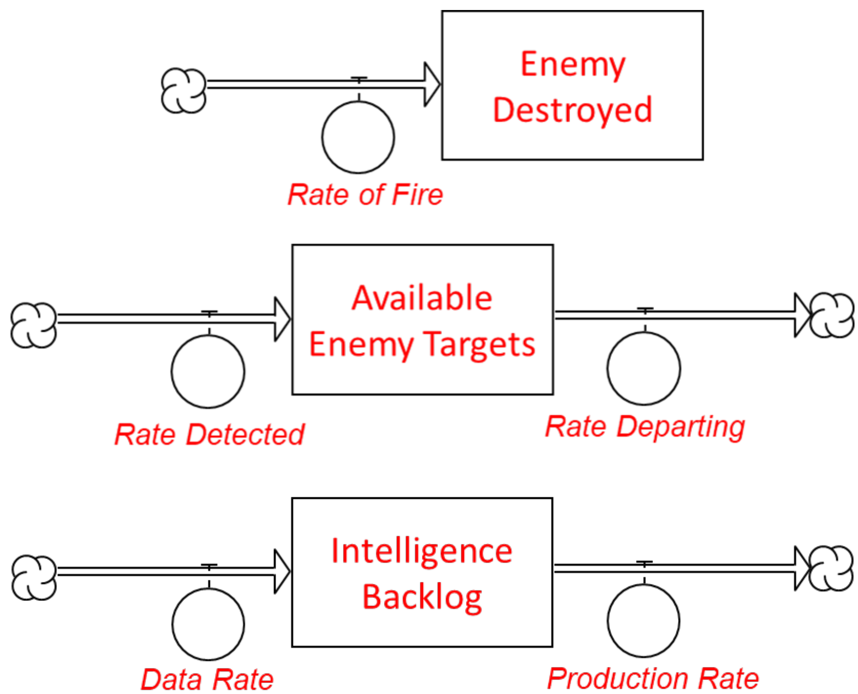


Figure 83: Use Case Stocks and Flows for Variables of Interest

The stock and flow diagrams will be combined with all causal feedback loops using commercial software. For this research, *Vensim* will be used due to its ability to quickly build SD models with the built in sensitivity analysis toolkits, built in Monte Carlo capability, and limited DES and ABM capability that can be included if necessary or if required for future development of the architecture.

An advantage of SD paradigm is that it operates at a global level, excluding extraneous details. By developing causal diagram, we can investigate the factors that contribute endogenously to the behavior of the system with respect to the desired MOP/MOE over time due to the structure of the system-of-systems. However, many parameters in an SD model may be difficult to obtain, while others may even be difficult to quantify and be subjective [141].

Using the three types of data sources, mathematical relationships between variables must be developed. For aspects for which data is unknown or too difficult to model, proxy variable will be used. Careful examination and SME input will help inform which 'knobs' (variables) that the decision-makers will be able to manipulate. This selection must be grounded in that which is technically feasible and will be initially aligned *with Roadmap 2017-2042* initiatives.

The next section within this chapter will discuss the creation of the model using available information; namely using the products developed previously using systems the mapping of DoDAF Operational Viewpoint models to create the M&S environment.

4.6 Apparatus/Simulation Design

Rather than depict AISR in a specific operational theater with a specific environment, the intent is to model the entire AISR PED enterprise from sensor to shooter, starting with theOV-1 (Figure 66) Enemy effects will be assumed. MDSS UAS assets are assumed to be operated remotely via MILSATCOM from sanctuary. The AISR PED enterprise will also include the newly proposed MDTF PED discussed in Section 0. Relationships not delineated in the DoDAF viewpoints will be constructed using published policy, SME expertise, and any available numerical data. Other values with be assumed as parametric analysis will be performed. In this study, we are primarily concerned with the enemy destroyed over time for which more is better; the ratio of enemy destroyed versus the friendly which a higher ratio is better, and intelligence backlog for which a lower value is preferred.

4.6.1 Modeling Friendly UAS and HIMARS

Using the information derived from the OV-1 and the Table 18 initial stock and flow diagrams with variable and connectors can be developed (see Figure 84). The initial stocks of interest are the UAS overhead, the satellites overhead and the pool of UAS available, as these are items that will decrease with time over the course of the operation and are directly related to the MOP/MOE. The flows can be deduced logically from the OV-1. The UAS overhead are depleted by enemy anti-aircraft fire and by the loss of UAS capacity; this can be caused by the loss of satellites needed to operate them, the return of UAS to base after a set period of time, and through the reduction of ground control stations or crews. These variables become converters that influence the flow rates or limit the value of the stocks. These relationships will continue to be refined with other DoDAF products and SME input via model testing.

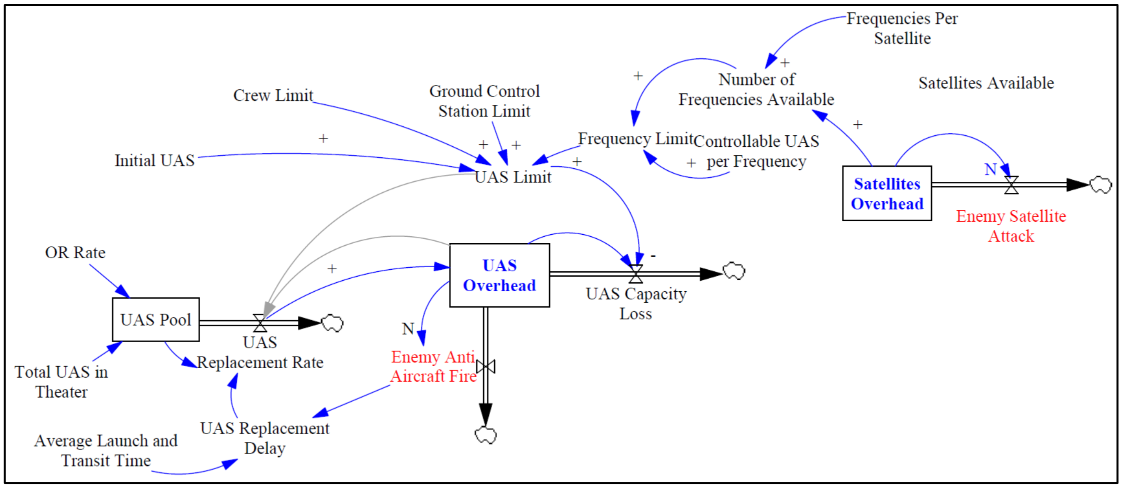


Figure 84: Initial Stock and Flow Diagram for UAS

Lastly, we know that the UAS in theater are a limited resource and, thus, must come from some reserve base. This available pool will be limited by the total UAS in theater, the operational readiness (OR) rate, and the number of aircraft required overhead. However, these UAS cannot be replaced immediately and will require additional delays to be added to account for average launch and transit times. While the OV-1 does not provide details as to the numeric values or mathematical relationships, the causalities are easily inferred. For example, if the OR increases then the available UAS in the UAS pool increases; such logic can be applied initially to all connections. From there, basic equations can be implemented. As discussed, SD equations are simple differential relationships. If more complicated relationships are required, it is customary to add additional variables in between two variables to keep relationships simple. This aids not only in the construct of the model, but also in the ability for stakeholders to easily see and understand relationships as they are simple to communicate with variable names in plain language [73]. The stock equation for the 'UAS Overhead' is depicted below in both differential and integral form. For a complete list of all equations used in the model see Appendix B.

$$\frac{d(UAS\ Overhead)}{dt} = UAS\ Replacement\ Rates(t) - UAS\ Capacity\ Loss(t) - Enemy\ Anti\ Aircraft\ Fire(t) \tag{5}$$

$$UAS\ Overhead(t) = \int_{t_0}^{t} [UAS\ Replacement\ Rate(s) - UAS\ Capacity\ Loss(s) - Enemy\ Anti\ Aircraft\ Fire(s)]ds + UAS\ Limit\ (t_0) \tag{6}$$

The 'UAS Overhead' has an initial value dictated by 'UAS Limit.' The number of aircraft that can be place overhead are limited by GCS, crews, and the frequency limit. The frequency limit is dictated by the variables associated with the satellites, including not only SME defined converters/variable associated with the satellites but also with the number of satellites as well. The number of satellites, in turn, is affected by enemy actions (cyber, shootdown, etc.). This trail of causes is depicted in the causal tree for the variable 'UAS Limit' shown in Figure 87. From just the simple OV-1 we can see complex relationships form; with changes in one affecting a key variable or changing a constraint/limitation. To account for this limitation on the 'UAS Overhead,' and outflow must be added as a function of these constraints that may change with respect to time.

$$UAS\ Capacity\ Loss(t) = IF\ THEN\ ELSE(UAS\ Limit <= UAS\ Overhead, PULSE\ (Time, TIME\ STEP) * (UAS\ Overhead - UAS\ Limit)/TIME\ STEP, 0\ ) \qquad (7)$$

The UAS Capacity Loss equation accomplishes two ends when determining 'UAS Capacity Loss' rate with respect to time. First the if/then/else statement calculates, at each time step, the difference between the 'UAS Limit' and the current 'UAS Overhead'. If the 'UAS Overhead' are greater than the organizations ability to support (due to loss of satellites, crews, etc.) then there is a corresponding drop in the number of 'UAS Overhead.' Dividing this difference by the time step serves two purposes. First, it puts the rate in the correct units of UAS per unit time. Second it applies the entire reduction instantaneously at the time step as pulse. The instantaneous pulse function is also known as the Dirac delta function $\delta(t)$, where $\delta(t)$, is the limit of a rectangular pulse starting at time 'T', with a width 'W' and a height of 1/W as the duration of the pulse goes to zero [51]. In mathematical terms it can be expressed as follows:

$$\partial(t, T) = \lim_{W \to 0} \delta(t, T, W) = \begin{cases} 0 \; for \; t \; \leq T \\ \dfrac{1}{W} for \; T < t \leq T + W \\ 0 \; for \; t > T + W \end{cases} \tag{8}$$

Figure 85 shows a standard rectangular pulse function over for rate of four UAS per hour starting at time T=10 hours for a duration of W=2.5 hours (shown in blue). The green line indicates the accumulation of UAS equal to 10 UAS.

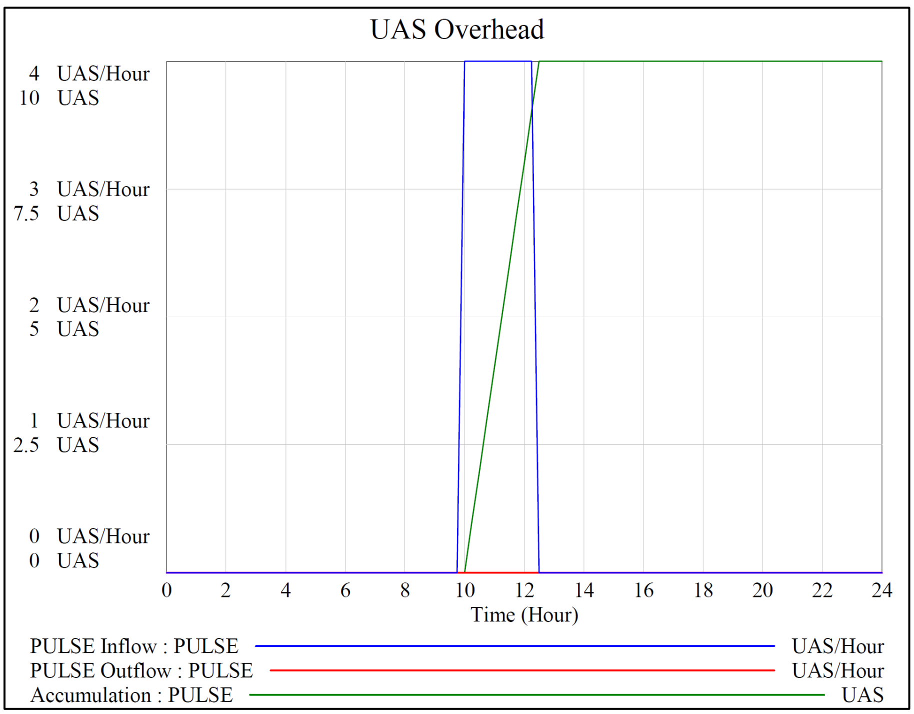


Figure 85: Standard Pulse Function Example

Note that the accumulation of UAS in this example is continuous, meaning at select times between 10- and 12.5-hours fractional values of UAS are included. By dividing the rate of UAS for the pulse by the time step (in this example 0.25 hours), and reducing the width of the pulse ‘W’ to the time step the increase of UAS is instantaneous at T=10 and the net accumulation is equal to the previous example of 10 UAS. The increase of the rate fourfold in this example is simply a byproduct of the technique to ensure the accumulation quantity is accurate without increasing the total UAS. In both examples, the accumulated UAS is10 UAS, but in Figure 86, the increase occurs from 10 to 10.25 hours. Decreasing the time step can cause this limit to approach 10, but with little added benefit to the model.

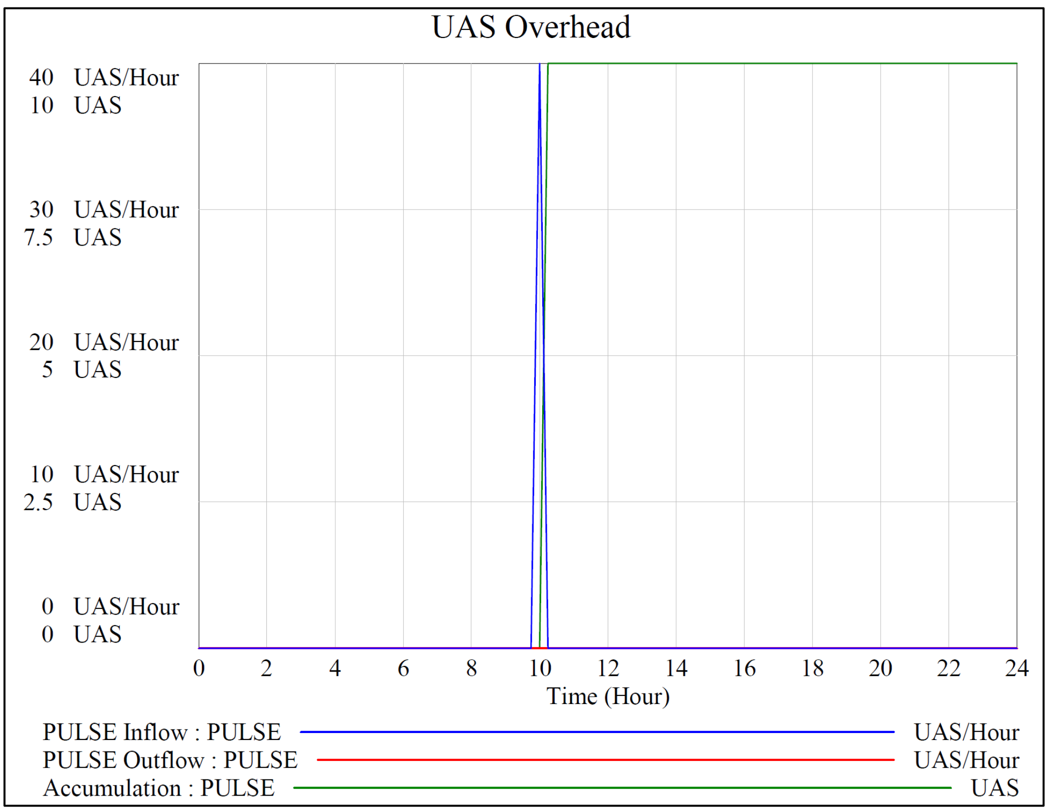


Figure 86: Dirac Delta Function Example

This is a method to add discretization to the continuous system for a more physical representation and better understanding for stakeholder [142]. As an example, there cannot physically be 9.73 UAS. While this is simply a function of the integration time step if a simple continuous rate is applied and can be explained as such, or could be justified as a degraded capability, it is more physically accurate to have the reduction of integer values of physical entities. However, for the purpose of this model for this dissertation, the Dirac delta function was utilized.

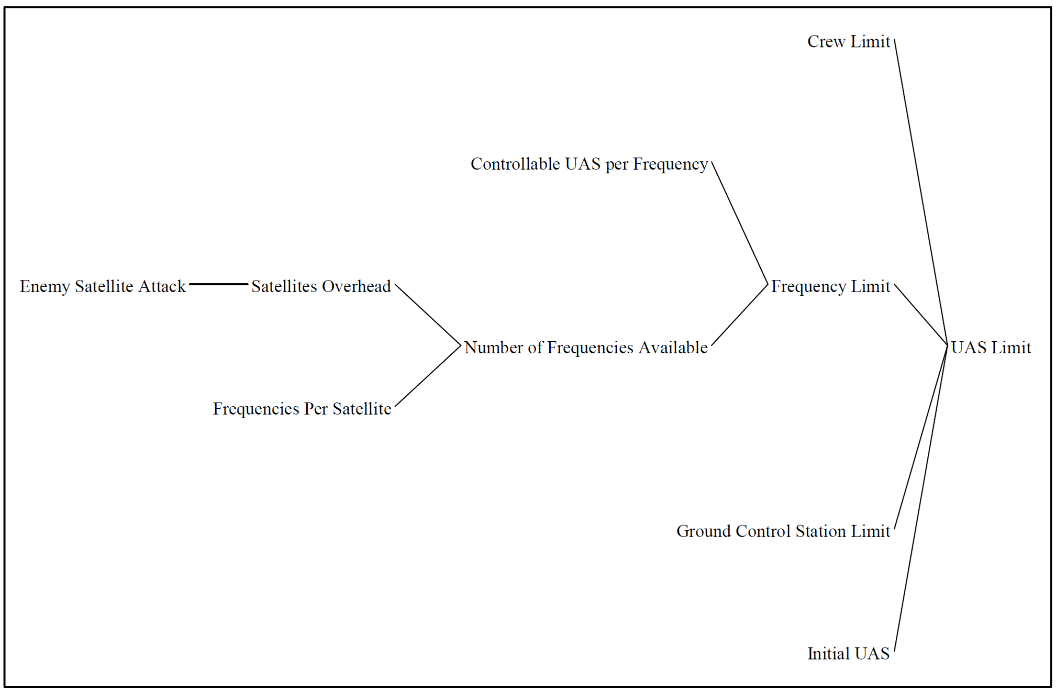


Figure 87: Initial UAS Limit Causal Tree

A similar equation is applied to the UAS replacement rate, that serves as both the rate of outflow from the 'UAS Pool' stock in reserve and the inflow to the 'UAS Overhead.'

$$
\begin{aligned}
UAS\ Replacement\ Rate(t) = IF\ THEN\ ELSE(\ &UAS\ Overhead \\
&< UAS\ Limit\ :AND\!:UAS\ Pool \\
&> 0\ , MIN(UAS\ Replacement\ Delay, UAS\ Pool \\
&/TIME\ STEP)\ , 0)
\end{aligned}
\tag{9}
$$

Like (8), (9) places qualifiers on the flowrate, giving it discrete behavior within a continuous model. The model is built under the assumption that for the initial ‘penetrate’ commanders require the maximum number of UAS Overhead as possible, limited only by the number of UAS available in theater and the capacity to operate them, dictated by UAS Limit. Hence, the if-then-else-and statement controls the outflow from the UAS Pool. If the UAS Overhead is less than the UAS Limit meaning there is additional capability bandwidth and there are still UAS available in reserve, then either a fixed integer value of UAS equal the value shot down by Enemy Anti-Aircraft Fire (see Figure 84) or the number of UAS remaining in the UAS Pool, whichever is less.

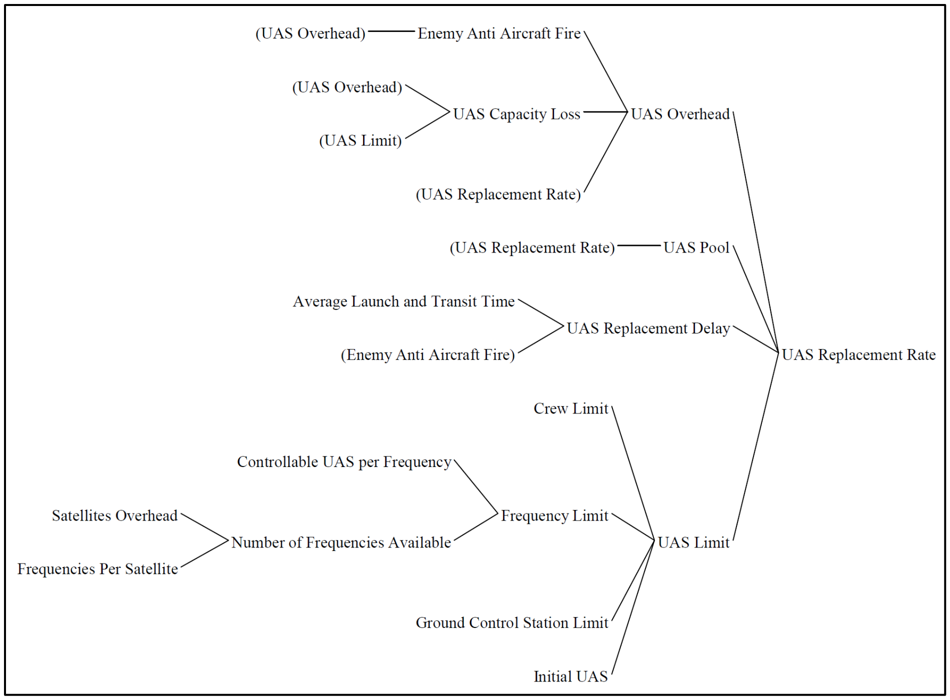


Figure 88: UAS Replacement Rate Causal Tree

Like previous outflows discussed, the UAS Replacement Rate value is divided by the time step to apply the effect immediately via the Dirac delta function rather than throughout the model's user-defined time scale (hours) to provide discrete additions. What differs the UAS Replacement Rate from other flows like UAS capacity loss or Enemy Anti-Aircraft Fire, is the inclusion of a time delay. Time delays are essential elements of SD and often the root cause of dynamic behavior in a system [51].

In SD modeling, a delay can be modeled in several ways, the simplest is through a series of intermediate stocks between uneven inflow and outflow rates where the outflow lags the inflow by some average time 'D'. this is known as a fixed-delay or pipeline delay and is commonly used for items or material in transit. A pipeline delay can be expressed as follows:

$$Material\ in\ Transit(t) = \int_{t_0}^{t} [Inflow(s) - Outflow(s)]ds + Material\ in\ Transit\ (t_0) \tag{10}$$

where:

$$Outflow(s) = Inflow\ (s - D)$$

n Vensim, this simple delay can be added using a basic “DELAY FIXED” function rather than having to add an intermediary stock and flow diagram, as is applied in the ‘UAS Replacement Delay’ function. The average delay duration is adjustable via the ‘Average Launch and Transit Time’ variable. It is important to note that the assumption in this basic delay type is that there is no mixing in order of the material in the process; items exit the accumulation in the same order they enter without consideration for individual items (recall SD works as aggregates). In a practical sense, this means the model does not account for specific UAS reaching its designated target area prior to another. All UAS are treated the same with an average launch and transit time to the target area. If specific aircraft to target area data is required to that granularity, rather than aggregate trends, then a different modeling type may be required. However, for top-level holistic analysis, average transit times can be used and made variable to gain valuable insights such as how increasing cruise speed (and decreasing average transit time), reducing launch times, or positioning UAS reserve assets closer to the target area (tactical considerations notwithstanding) could affect the successful completion of overall operational MOP/MOE. This can be done simply by providing a parametric range to 'Average Launch and Transit Time' for the ability to conduct univariate or multivariate sensitivity testing which will be discussed later. If this average transit time is found to be of considerable influence, then higher fidelity modeling may be required.

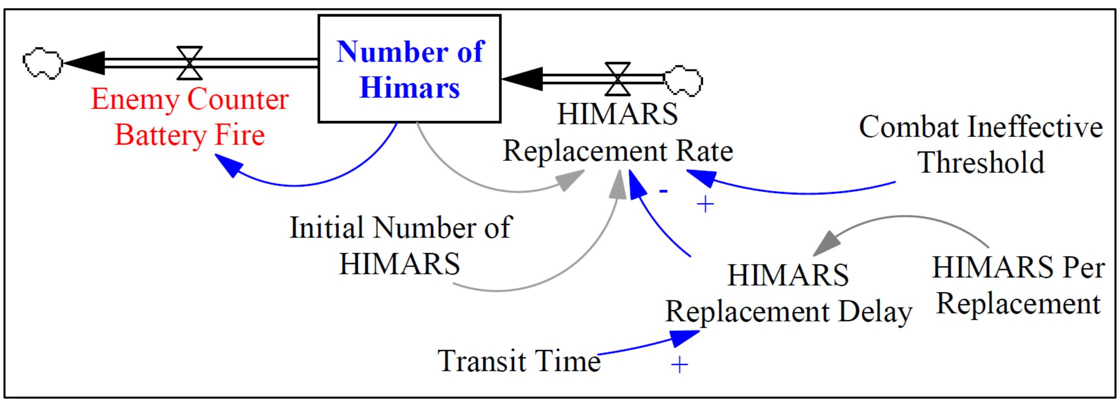


Figure 89 Initial HIMARS Subsystem Stock and Flow Model

Referring to the OV-1, it can be observed that the number of HIMARS is also considered an important variable interest during combat operations, hence it a stock. The outflows are the rate at which HIMARS destroyed by enemy counter-battery fire and flow in is due to replacements. Like with UAS, other converters and connectors can be inferred, specifically those influencing the replacement rate (inflow). Like the UAS, there is a delay prior to replacement of the HIMARS due to transit time. Additionally, the addition of the variable "HIMARS per Replacement" allows the user to adjust the number of HIMARS that are replaced with each Dirac delta function pulse. What differs between the 'UAS Replacement Rate' and 'HIMARS Replacement Rate' is a matter of modeling assumption based on interviews with SMEs and doctrine. While UAS are tasked individual to NAIs, HIMARS are deployed in organizational unit structures (battalions, batteries, etc.) and would be employed as such. It is assumed in the model that the entirety of the MDTF Brigade's HIMARS battalion would be employed from the onset of hostilities. Only after a certain friendly loss threshold ('Combat Ineffective Threshold') would reserve units be called in if available. If no reserve units are available or are excluded for the analysis, either the 'Combat Ineffective Threshold' or 'HIMARS Replacement Rate' can be set to zero, effectively shutting off the inflow valve (see below).

$$\begin{aligned}
&HIMARS\ Replacement\ Rate(t) \\
&= IF\ THEN\ ELSE(Number\ of\ Himars \\
&< Combat\ Ineffective\ Threshold \\
&* Initial\ Number\ of\ HIMARS, HIMARS\ Replacement\ Delay \\
&/TIME\ STEP, 0)
\end{aligned} \tag{11}$$

Like the UAS, enemy attacks that affect the outflow rate of the HIMARS are a challenging but extremely critical unknown. For the initial assessments, as is typical in SD, an assumed average rate can be assumed and given a parametric range. By providing a simple assumption initially, it allows for easier comparison of effects. However, because SD is continuous the rate will be integrated over the simulation's time step interval. This means that the number of HIMARS or UAS will decrease continuously as well, which can be confusing to the user as at intermediate time periods, as there will be fractional numbers of HIMARS or UAS as discussed previously. Therefore, a discrete Dirac delta function is again applied for the outflows of UAS Overhead, Number of HIMARS, and Satellites Overhead. However, the frequencies and magnitude of the enemy fire has to be taken into account as a repeated event.

Regardless of how many DoDAF diagrams are used, none can account for enemy action and the enemy effects remain a virtual unknown. If modeling a historic engagement, post battle data can be used to generate a simulation, but the simulation will only be correct for the past engagement. In short, the actions and effects of the enemy anti-aircraft artillery and counterbattery artillery will always be an unknown and for all intents and purposes, be random. A smaller refined engagement model that incorporates terrain (such as an agent based model) can be used for the engagement information and after thousands of runs, surrogate models can be created and inserted into the larger holistic SD model as has been done for other systems [100, 105]. However, that hybrid model integration is beyond the scope of this thesis. Additionally, such a method is time consuming, while this effort is meant to be applied rapidly during CBA generation of the ICD. Therefore, the implementation of a stochastic function can be applied to generate random effects of enemy action for initial analysis.

The Vensim™ software utilized in this study contains several random number functions. Because we desire a discrete integer value for the outflow representing the destruction of these friendly assets, we require a discrete probability random number function of which two are available in Vensim™. The first, a binomial distribution seems plausible because an enemy attack is either a hit or a miss. However, in Vensim this random probability is assessed at every time step. Therefore, if a simulation is conducted to replicate 24-hours with a step time of 0.125 hours, it will assess strikes as successful hits or misses 192 times in the 24-hour period. If even half are assessed as hits, our HIMARS and UAS would be eliminated in the first few hours of the engagement. Such a result has the potential to be useful in a force on force warfighting simulation but is of little to no value when conducting analysis of technology or allocation alternatives.

The alternative random number function that utilizes a discrete probability distribution available in Vensim is the 'Random Poisson' function which utilizes the Poisson distribution defined as follows:

$$f(x) = P(X = x) = \frac{e^{-\lambda}\lambda^{x}}{x!} \; for \; x = 1,2,3, \ldots \tag{12}$$

Where

$$\mu = E(X) = \lambda \text{ and } \sigma^2 = Var(X) = \lambda \text{ and } \sigma = SD(X) = \sqrt{\lambda}$$

The Poisson distribution expresses the probability, given a known constant mean rate, that a discrete number of events will occur in a fixed interval of time or space. Each event is independent of previous events.

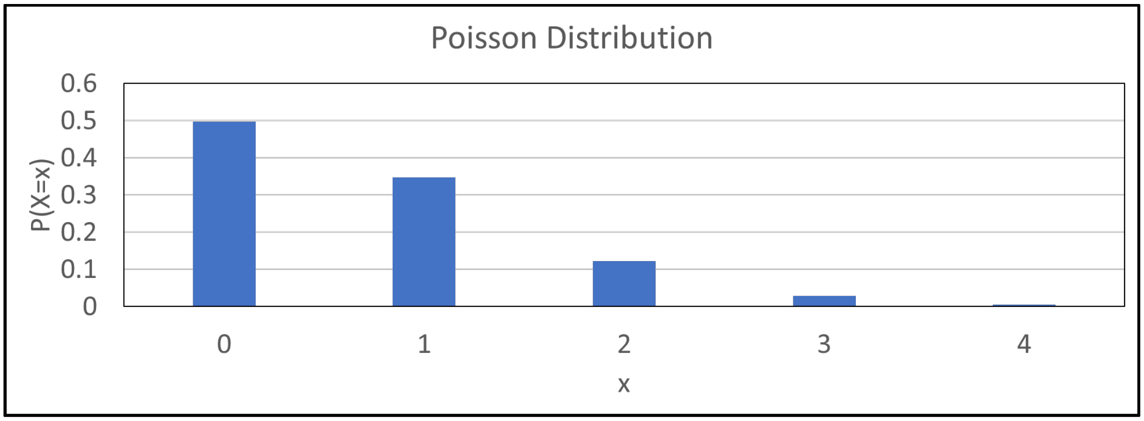


Figure 90: Example Poisson Probability Distribution λ=0.7

This function is appropriate for the targeting of friendly UAS and HIMARS that are assumed to not be stationary and hence firing solutions for each engagement are independent. In the SD model, the Vensim function RANDOM POISSON(m,x,M,h,r,s) creates a Poisson distribution with a minimum value of ‘m’, the maximum value of ‘x’, a mean value ‘M’, a shift parameter ‘h’, a stretch parameter ‘r’, and a stream ID ‘s’. The stream ID identifies an independent random number stream to use that is reproducible so that each run generates the same results [142]. The stream ID can be varied by introducing a ‘NOISE SEED’ variable that allows the user to change the stream ID to observe different random number strings and evaluate the effects of changing enemy actions (outflows) on the dynamics of the system against desired MOP/MOE.

Such a characteristic is an important when evaluating different alternatives for technological, structural, and policy implementation; just because a solution shows effective against a stream of outcomes, does not mean that is suitable against others. It can be assumed that rate at which enemy actions that destroy friendly assets will have a considerable effect on the performance of the system-potentially more so than anything else. Therefore, solutions and combinations of user-controlled variable for the system-of-systems must be robust. How to address this robustness against noise using surrogate models is addressed in the subsequent chapter. However, since enemy actions and effects are so uncertain, given the same periodicity for enemy effects, initial holistic evaluation can still provide effective comparison or which policies perform better given the same noise seed and exact same likelihood of occurrence, afterward solutions can be evaluated against different noise streams.

The application of the Dirac delta function on asset outflow due to enemy action with the RANDOM POISSON function can be seen in the following two figures. Each has a mean of 0.7 so that there is a higher probability of discrete asset outflow assessed with $P(X=0)=0.496$; $P(X=1)=0.347$; $P(X=2)=0.12$; $P(X=3)=0.028$; with the $P(X>3)=.006$. The first, Figure 91, uses noise seed=0, while the second, Figure 92 uses noise seed=5.

Note that Dirac delta function combined with the Random Poisson provides corresponding discrete reductions in the number of HIMARS (recall that the reduction value is divided by the time step, in this case 0.25 hours so the rate is four times the magnitude of the physical reduction). Note that with a noise seed of 0, the frequency and amplitudes of the outflows are different than those with a noise seed of 5. As a result, while the exponential decline trends are the same, there are fewer HIMARS remaining with noise seed 5.

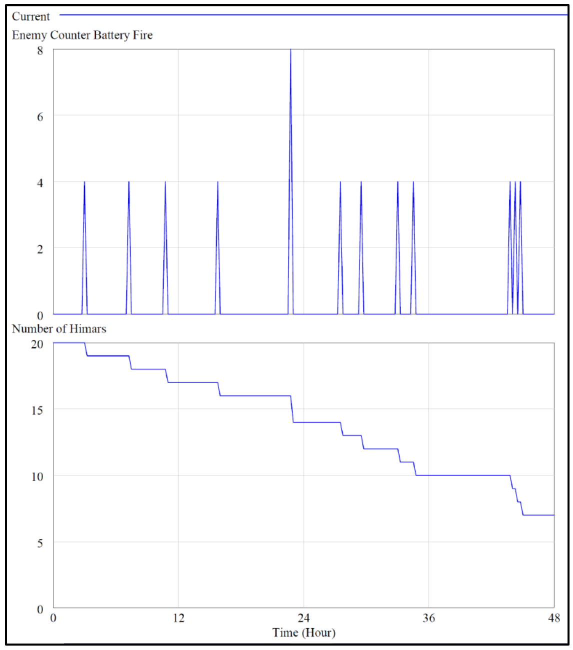


Figure 91: Example Enemy Strikes on HIMARS Noise Seed 0

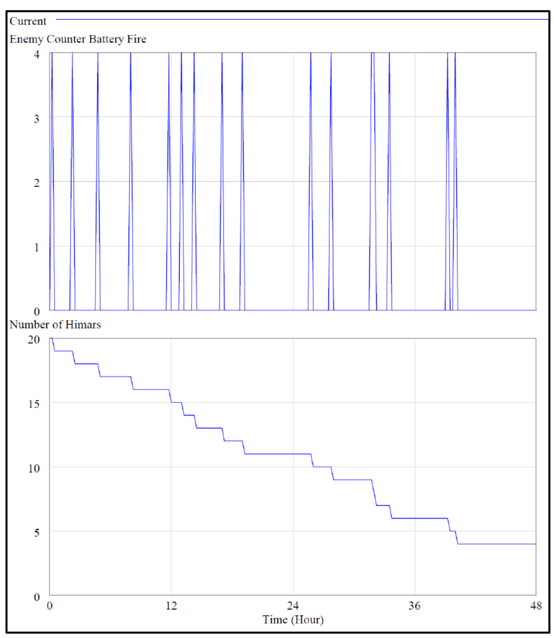


Figure 92: Example Enemy Strikes on HIMARS Noise Seed 5

Additional realism can be added through logic rather than force-on-force simulation by means of additional variables/converters to account for disparity of forces, though not for the effects of maneuver or terrain. It is logical to assume that as the number of enemy counter battery artillery and AAA assets decrease due to HIMARS kinetic strikes, that there will, in turn, be fewer strikes against the friendly HIMARS and UAS. Additionally, it can be assumed that as friendly HIMARS are killed and the numbers are reduced, the likelihood of hitting the remaining assets decreases as fewer targets remain. Therefore, by introducing additional converters, these variables can be multiplied by a constant to create a variable mean value (λ) for the Poisson distribution that decreases as enemy and friendly assets are reduced.

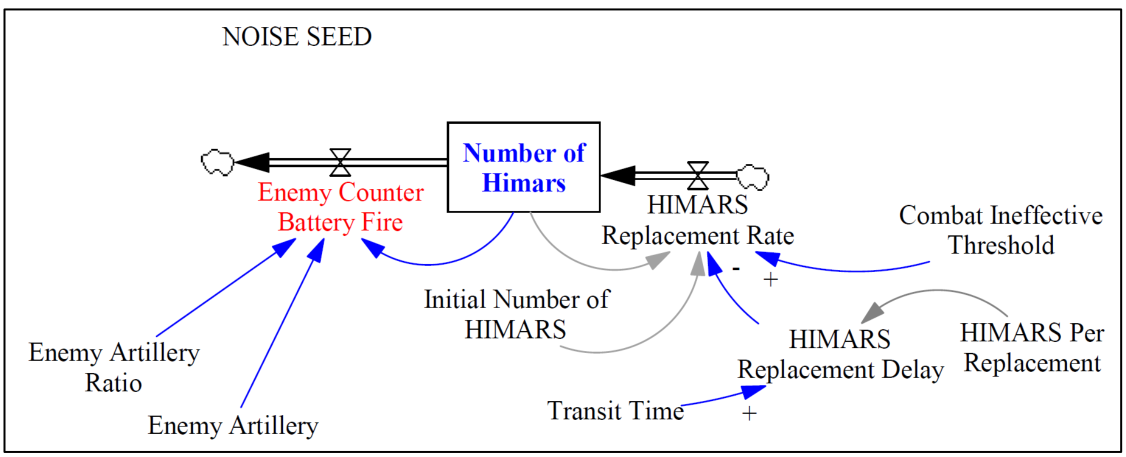


Figure 93: Updated HIMARS Stock and Flow with Variable Mean Outflow

An updated version of the HIMARS and stock and flow diagram that incorporates these variables is shown in Figure 93 with its corresponding conditional Enemy Counter Battery Fire outflow equation shown below.

$$
\begin{aligned}
&Enemy\ Counter\ Battery\ Fire(t) \\
&= IF\ THEN\ ELSE\ (Enemy\ Artillery > \\
&= 1{:}\,AND{:}\,Number\ of\ Himars \\
&>\ 0, RANDOM\ POISSON(\ 0, Number\ of\ Himars, 0.007 \\
&*\ Number\ of\ Himars \\
&*\ Enemy\ Artillery\ Ratio\ , 0\ , 1\ , 0)/TIME\ STEP\ , 0)
\end{aligned} \tag{13}
$$

This equation sets conditions that are easily defined and necessary to prevent negative flow; there must be at least one Enemy Artillery and one friendly HIMARS for outflow to continue. Note that the equation also incorporates IFTHENELSE qualifiers so that if the enemy or friendly assets are eliminated the outflow goes to zero and prevents negative stock (you cannot have negative assets. There are many methods to do this, often done with average rates multiplied by the stock. MAX MIN and IF THEN ELSE are also acceptable methods, but, like many modelling efforts, the best method is a matter of model preference, situationally dependent, and a subject of debate in the field.

Likewise, the UAS Overhead model can be improved to account for Enemy Anti-Aircraft effects. Also, additional converters can be added to expand on the original converters in Figure 84. These converters represent physical entities or policy considerations that stakeholder and decision makers can affect.

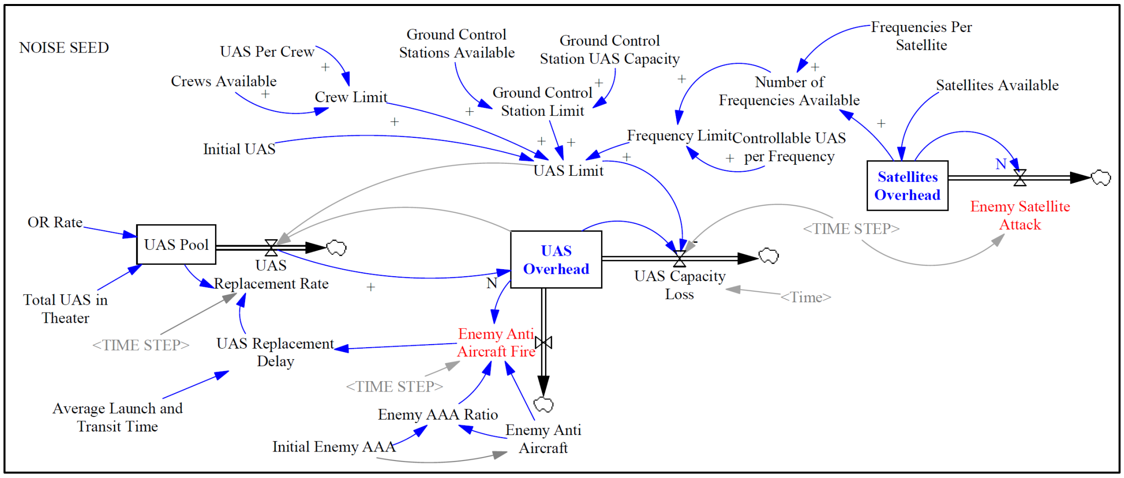


Figure 94: Updated UAS Overhead Stock and Flow Diagram

Like the Number of HIMARS, the UAS Overhead behavior is also influenced by enemy actions, and like the Number of HIMARS, the UAS Overhead leverages a Dirac delta function with a RANDOM Poisson with a shifting mean as UAS Overhead and Enemy AAA decrease. With the added converters, a more complex causality tree is formed (Figure 95) with an added layer over the initial causality tree for the UAS Limit(Figure 87).

From the stock and flow diagram and the causal tree, the dependencies become apparent. Unlike the Number of HIMARS, the UAS is dependent not only on the available stock in reserve, but also on the number of satellites overhead as an additional stock that affects the outflow. Before combining this subsystem of the model with others, it is important to test behavioral relationships and assess behavioral trends. Model testing and validation will be conducted first for subsystems connections then for the entire system as recommended by Ford [143]. This process is iterative. Model testing and validation types and results should be documented thoroughly along with any constraints and assumption made in the process.

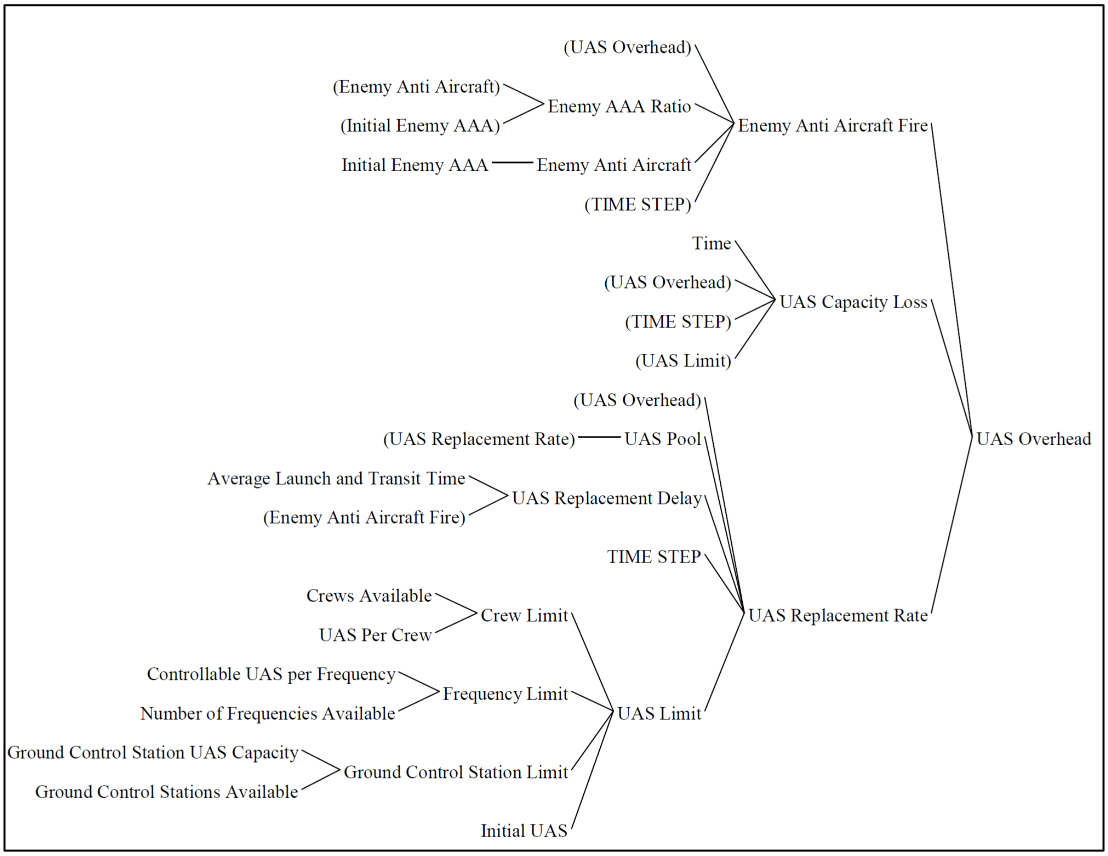


Figure 95 Updated UAS Overhead Causality Tree

4.6.1.1 Validate and Verify the Model

According to Dr. Christopher Chung, associate professor in the Department of Industrial Engineering at the University of Houston there are two major types of validation for models and simulation. The first being face validity and the second being statistical validity. Face validity is straightforward and literally means that, on its face value, the model represents reality which is typically done with the assistance of SMEs knowledgeable on the system under study. This approach instills a sense of ownership and helps avoid last-minute inquiries into model elements. However, this also depends on the perspective and interpretation of the SME; a model may be valid to one person but not to another. For face validation, it is necessary for the modeler to ensure there is enough animation of the process with enough visual fidelity of the actual process to assuage the SMEs and key stakeholders.

Statistical validity requires an objective quantitative comparison between the outputs of the actual system and the model. The smaller the statistical difference between the real-world and model/simulation data sets, the more valid the model.[79] However, this requires data to be available in the first place. In the real world, there may nothing by which to compare the model, such as in instances of proposed real-world systems. Furthermore, while a model or simulation may be statistically valid for a model or simulation at the onset, this does not guarantee validity when the model is changed to explore alternative methods or layouts of a system.[81]

Dr. Stewart Robinson, of the Operations Research and Systems Group at the University of Warwick, posits that there are six types of validation for models: conceptual model validation, data validation, white-box validation, black-box validation, experimentation validation, and solution validation. Conceptual model validation, like face validation, in ensuring that the conceptual model contains the necessary scope and level of details necessary to meet the objective of the study.

Data validation involves determine what data is necessary for model construction and validation, and that this data is sufficiently accurate. White-box validation is assessing if the constituent parts of the computer model accurately represent the real-world elements they are intended to. Black-box validation is the macro check of the model's operation. White-box and Black-box validation are frequently collated under the term 'operational validity.' Experimental validation is the act of ascertaining the accuracy of the experimental procedures that will be conducted with the M&S. Lastly, solution validation, like statistical validation assess the accuracy of the outputs of the model. This process is iterative over the life-cycle of the simulation study.[81]

The aim of V&V is to confirm that a model is sufficiently accurate for the purpose for which it had been built and for which it will be used. A common misconception of V&V is that it proves a model is *correct*. In reality, the intent of V&V is to attempt to prove the model is *incorrect* and by failing to do so build the user and the modeler's confidence in the model and its results.[81] Even "painstaking constructed models may not actually represent reality."[79] As famed statistician, George E.P. Box once stated that "all models are approximations. Essentially, all models are wrong, but some are useful. However, the approximate nature of the model must always be borne in mind."[144]

It is imperative to know the limitations, assumptions, and constraints of the models being used; there are simply too many variables and uncertainties in military operations, let alone in war. The *Code of Best Practice: Experimentation* notes that while models and simulation can be used to test limits of experiments, but are "neither valid nor robust enough to explore the range of conditions relevant to military operations."[145] That being said, a program still has to go through the verification and validation processes before experimentation can begin; this process is iterative.

4.6.1.2 Model Testing

Model testing is the means of conducting verification and validation and is therefore often abbreviated as VV&T. This is often done be subjecting the model to test cases or data to ensure it is functioning properly by demonstrating inaccuracies or revealing errors.[76] During formal model testing there are two types of structured tests, 'direct structure tests' and 'structure-oriented behavior tests'.[146] ***Direct structure*** tests involve assessing each of the relationships individually in the model to the real system (mathematically or logically). These direct structure tests can be conducted empirically or theoretically. However, just because individual relationships are correct it does not ensure that the holistic model is correct. ***Structure-oriented behavior tests*** involve running the simulation for the entire model to uncover potential structural flaws. Several such tests exists such as extreme-condition/stress (indirect) tests, behavioral sensitivity tests, and phase-relationship tests.[146]

The model and simulation developer should be wary of customers who seek validation of their predetermined notions about the system behavior, as they will likely not accept results that are contrary to them. Like Forrester, Sterman posits that validation and verification are impossible, though as discussed falsification of the model is possible. If one cannot prove the model false, then it is acceptable for the given problem.

For actual DoD M&S to be used, DoD Policy requires that it be verified validated and accredited. Per DoDI 5000.61, *DoD Modeling and Simulation (M&S) Verification, Validation and Accreditation (VV&A)*, "Models and simulations used to support major DoD decision-making organizations and processes …shall be accredited for that specific purpose by the DoD Component M&S Application Sponsor."[26] VV&A should be completed during the development efforts of the M&S to reduce associated costs. DoD policy requires the cost of implementing VV&A be commensurate with the risk associated with the criticality of the decision being informed by the M&S. M&S decisions that could result in the loss of life obviously warrant more thorough V&V while some M&S simply require, at minimum, sensitivity analysis be performed to identify the elements that are most influential and require more through focus to prevent a misleading result.[26]

Therefore, the purpose of this thesis is not to develop a fully verified, validated, and accredited model of the AISR-PED architecture, rather to demonstrate a methodology by which to model and answer the questions of interest that could be more thoroughly undergo DoD VV&A.

Table 18 lists model testing methods recommended by Sterman [51]. Those that were able to be conducted for the creation of Validation will align with the position of both Sterman and Robinson. Namely, the model cannot be validated by historical data or otherwise. Rather, an attempts were made to invalidate the model through model testing; inability to invalidate the model along with having constructed it with the SME customer builds confidence that the model is a reasonable depiction of future behavior based on variation of input variables.

Table 18: Model Testing Methods

Test	Purpose	Procedure	Y/N
Boundary Adequacy	Check if important concepts to address problem are endogenous.	Use generated products (boundary charts, etc.), consult SMEs	Y
Structure Assessment	Check appropriate aggregation and description of system of interest	Use policy diagrams (DODAF) and SMEs	Y
Dimensional Consistency	Ensure equations dimensionally consistent	Dimensional analysis software in Vensim	Y
Parameter Assessment	Check if parameters have real world counterparts	Statistical methods, modal tests, SMEs	Y
Extreme Conditions	Check if model response is plausible	Univariate and multivariate extreme value inputs	Y
Integration Error	Check sensitivity to time step choice	Vary time steps and observe	Y
Behavior Reproduction	Check if generate modes reflect historical real behavior	Statistical measures of correspondence with real system (R^2, MFE, MAE)	N
Behavior Anomaly	How does variation of assumptions change behavior?	Replace Assumptions	Y
Family Member	Calibrate model	Compare against other models of same system	N
Surprise Behavior	Check for previously unobserved or unrecognized behavior	Accurate documentation of simulation runs	Y
Sensitivity Analysis	Check numeric, behavioral and policy sensitivity	Initially through built-in sensitivity tools	Y
System Improvement	Check if model improved system	Compare to baseline or control	Y

Because a subsystem for enemy actions and friendly effects on enemy numbers has yet to be developed for the holistic system, an enemy reduction rate can be assumed to test the subsystem model. For testing of this subsystem, Initial Enemy AAA is assumed to be 75 and are reduced by 1 every hour for a 48-hour run. The top graph in Figure 97 Subsystem Test UAS Overhead Causal Graphs, shows the behavior of the UAS Overhead through the first 48 hours. The number of UAS Overhead is initially capped at 20 due to the UAS capacity limit in Figure 98 despite a desired initial value of 30 UAS overhead. Initially, crew limit, GCS limit, and frequency limit are all set at 20 to test if the frequency limit reductions dictate UAS limit. The frequency limit is reduced in correlation to the loss of friendly satellites due to enemy action (Figure 99). Returning to the UAS Overhead, dynamic oscillations are observed for the first 12 hours due to a balancing loop with delay (see Figure 44) between Enemy Anti-Aircraft Fire, UAS Replacement Rate and UAS Replacement Delay. (See Figure 96). This behavior is followed by exponential decline (if smoothed) due to the exhaustion of UAS Pool reserves. There are two additional observations of note germane to the testing of the model construct. First, as intended, as UAS Overhead decreased and Enemy AAA decreased, the mean of the RANDOM POISSON function shifted toward zero, decreasing the likelihood of Enemy Anti-Aircraft Fire demonstrated by the reduced frequency of the discrete outflow (Figure 97).

Loop Number 1 of length 1
UAS Overhead
UAS Replacement Rate
Loop Number 2 of length 1
UAS Overhead
UAS Capacity Loss
Loop Number 3 of length 1
UAS Overhead
Enemy Anti Aircraft Fire
Loop Number 4 of length 3
UAS Overhead
Enemy Anti Aircraft Fire
UAS Replacement Delay
UAS Replacement Rate

Figure 96: UAS Overhead Loop Mapping

Second, the UAS Capacity Loss is 0 over the duration (see Figure 97 third graph from top) despite continued decrease of the UAS Limit due to decreased Frequency Limit (see Figure 98 first and third graphs) due to loss of Satellites Overhead caused by Enemy Satellite attacks (Figure 99). This is not a flaw in the M&S, on the contrary. The UAS Capacity Loss remains at zero because the rate at which UAS Overhead are decreasing due to Enemy AAA is greater and, hence, the reduction in Satellites Overhead does not influence the number of UAS. This assertion can be further tested by adjusting the outflow from the UAS Overhead due to Enemy Anti-Aircraft fire while keeping all other variable setting the same. Figure 101 compares the runs with and without Enemy Anti-Aircraft and demonstrates the desired effects of UAS Capability Loss. Note that despite the loss of UAS Overhead, no replacement UAS are sent from the UAS Pool due to lack of capacity to control them which is the correct behavior that is desired (Figure 101).

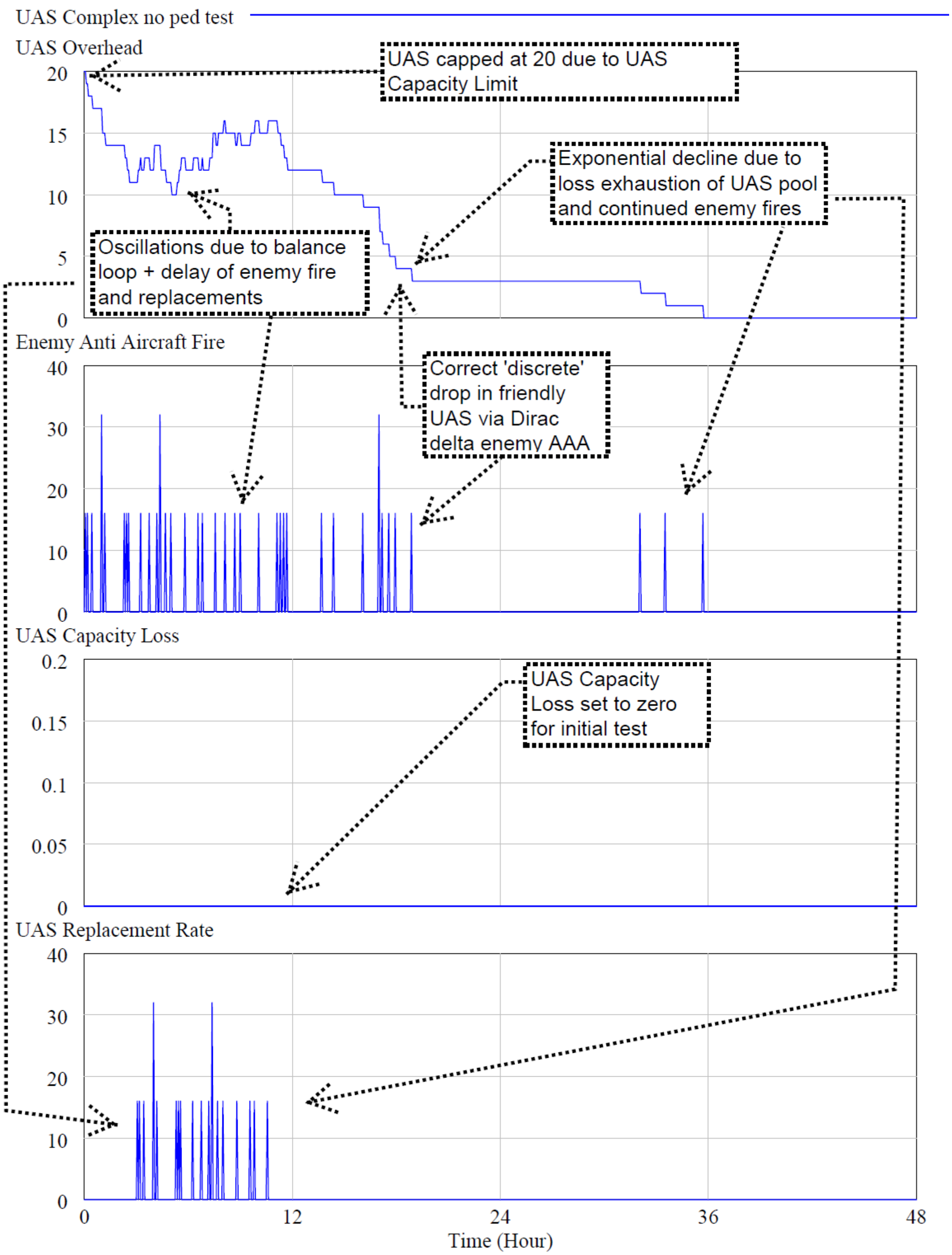


Figure 97 Subsystem Test UAS Overhead Causal Graphs

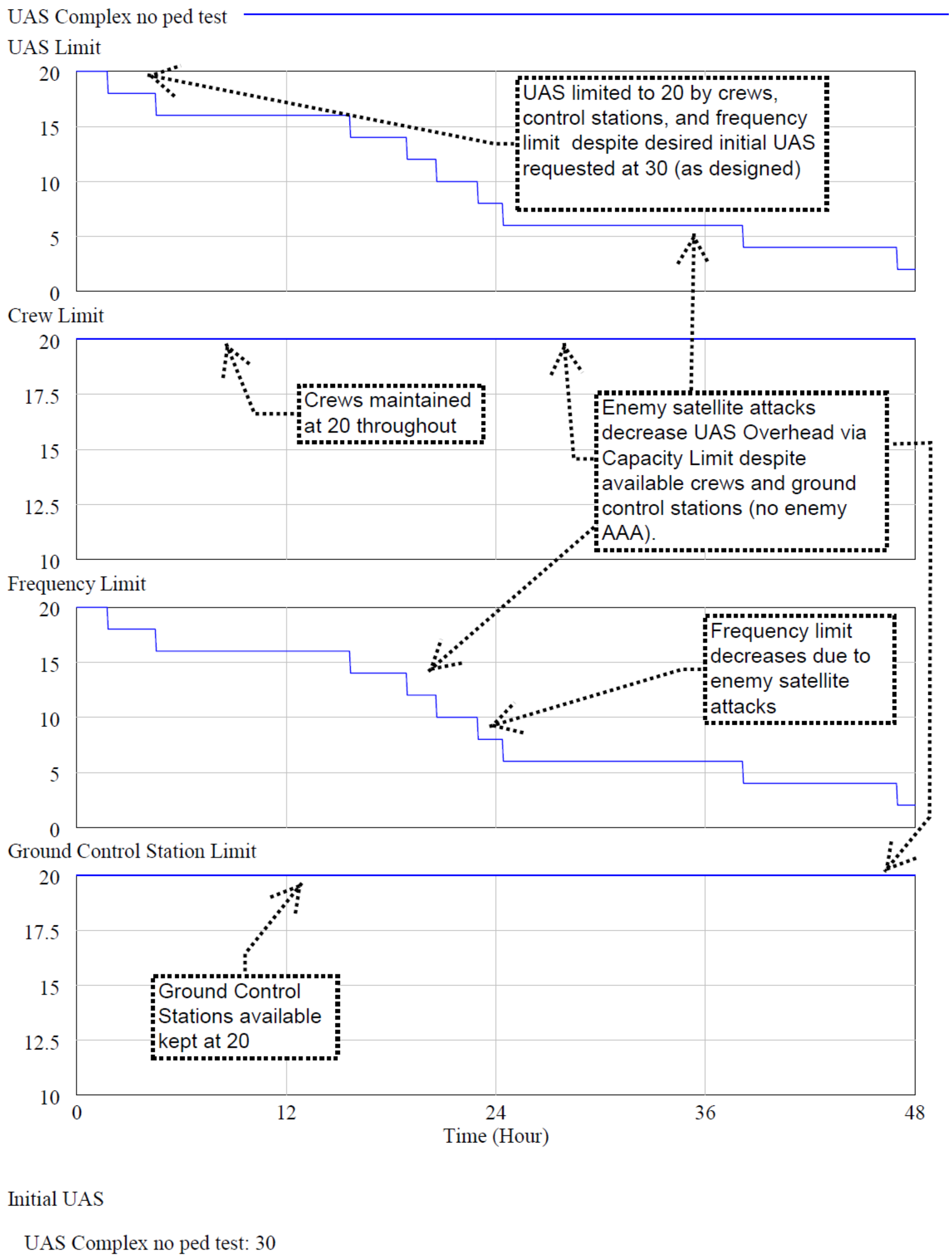


Figure 98: Subsystem Test UAS Limit Causal Graphs

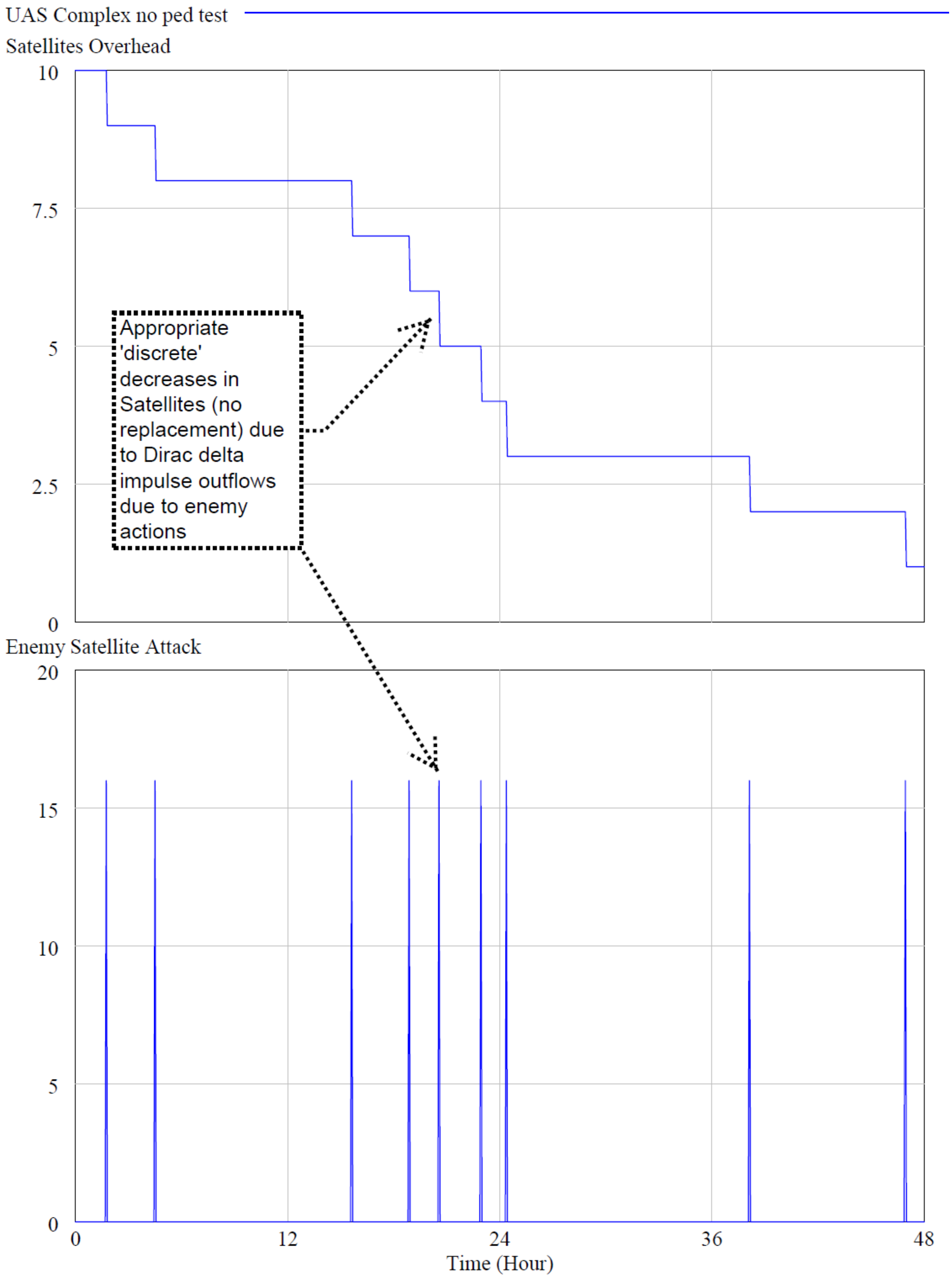


Figure 99: Subsystem Test Satellites Overhead Causal Graphs

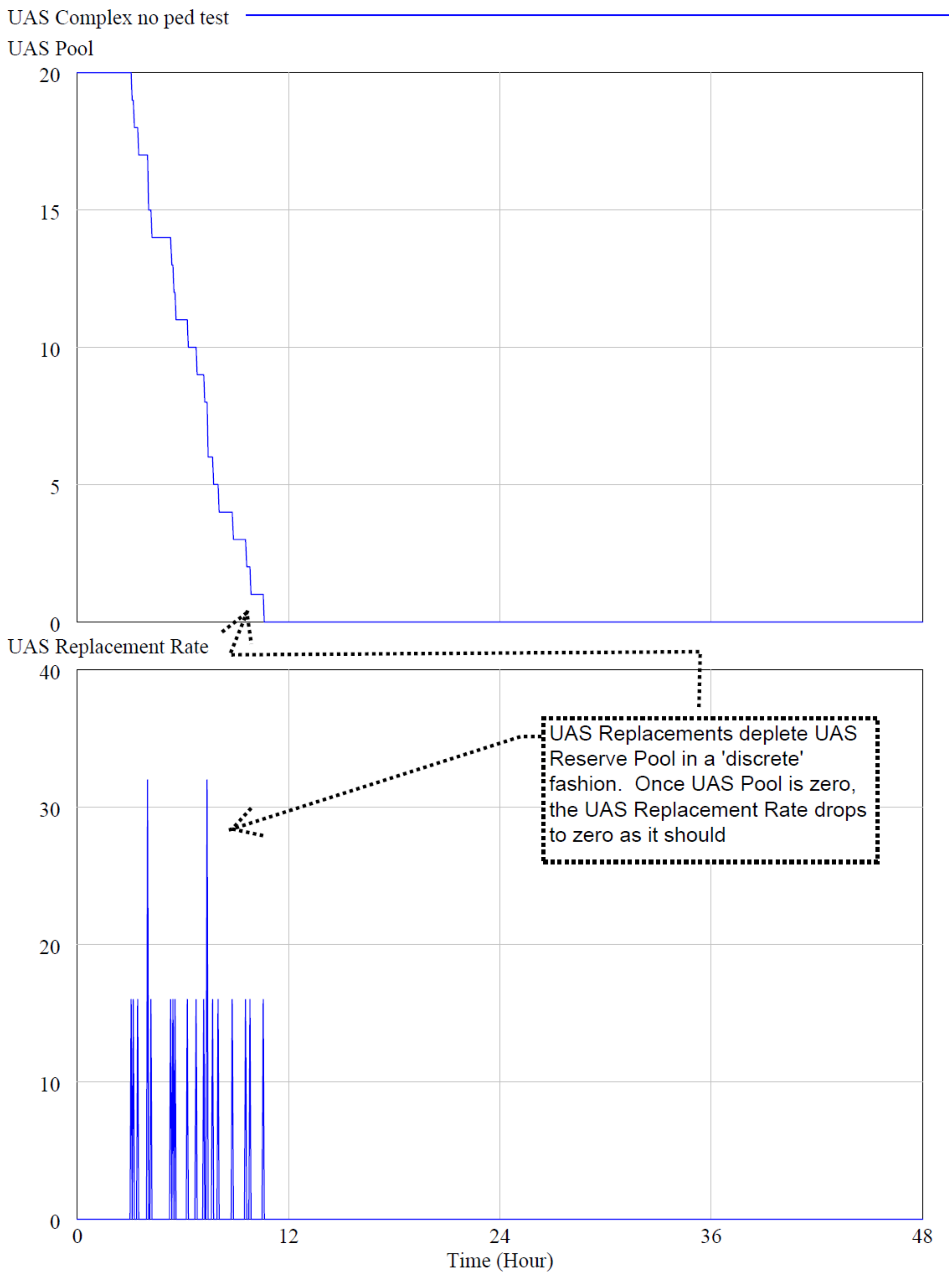


Figure 100: Subsystem Test UAS Pool Causal Graphs

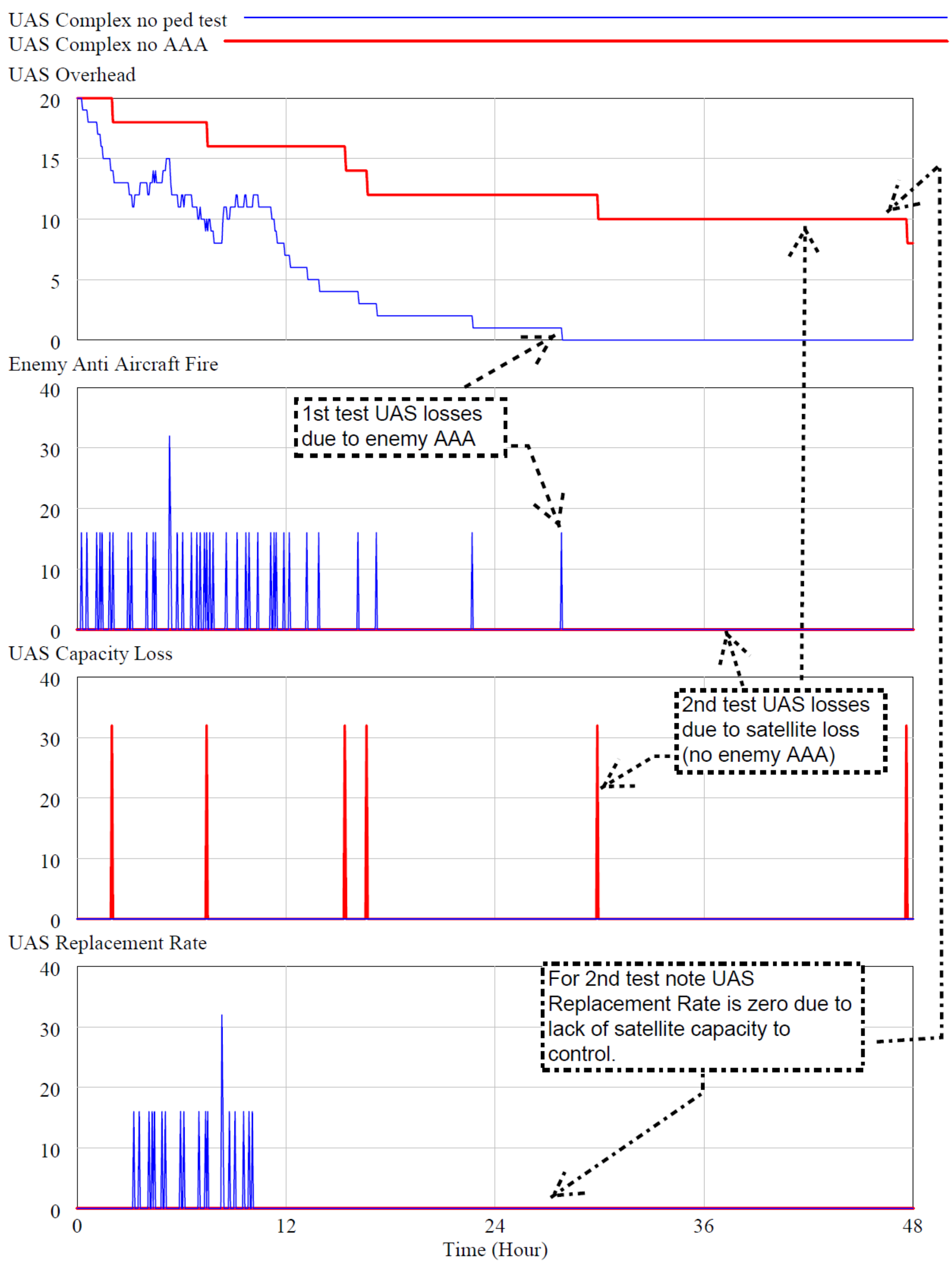


Figure 101: Subsystem Test UAS Overhead without Anti-Aircraft Comparison

These basic sensitivity tests as well as dimensional analysis were repeated for every subsystem by univariate testing to ensure no abnormal behavior, particularly at extreme value ranges. Additionally, testing was conducted using manipulation of two or more influential values at extremes to ensure reasonable behavior for the range of values applied, using SME feedback to refine relationships and assess initial behavior nodes. By following Fords method, this tests that the subsystem models cannot be invalidated and hence are valid moving forward in absence of historical data. This could be improved if additional data is available, but for combat scenarios, such data is unlikely. In the interest of brevity these subsystem graphs will not be repeated. However, as Ford [143] notes, testing the subsystems individually does not guarantee the combined system-of-systems model will accurately depict behavior. Hence, it will be necessary to again test the combined system in this manner as will be addressed in the subsequent chapter.

4.6.2 Modeling Enemy Assets

The next subsystem of interest is the enemy assets (Anti-Aircraft Artillery and Enemy Artillery). These changing stock values are important metrics for the operational as defined in Table 11: AISR-PED Measures of Performance and Table 12: AISR PED Measures of Effectiveness. Obviously, there are no DoDAF models for enemy assets beyond the OV-1 that depicts enemy assets of interest. Therefore, the initial enemy stock and flow diagrams are relatively simple; both Enemy Anti-Aircraft and Enemy Artillery are each a stock with the rate of replacement as an inflow and a destruction rate as the outflow. As a bookkeeping measure, the outflow of each of the enemy stock times becomes an inflow for a stock representing the total cumulative enemy destroyed. It is in the outflow or destruction rate that represents a modeling challenge. From the DoDAF, SME knowledge, and the application of a bit of logic using causal diagrams, the initial converters and connectors can be generated by asking the simple question: what can influence the destruction rate? More importantly, what variables can stakeholders/decisionmakers control that have an influence on the effects on target? Logically, there are a limited number of HIMARS and those HIMARS can only target one thing at a time, how those fires are directed will influence the destruction rates; the more HIMARS and their focus of fires will increase the destruction rates accordingly. Furthermore, to target the enemy, they must first be detected. The detection rate and subsequently the targeting rates and hence the destruction rates reasonably increase as the number of UAS Overhead increase. From these simple relationships, we can construct the initial stock and flow diagram in Figure 102.

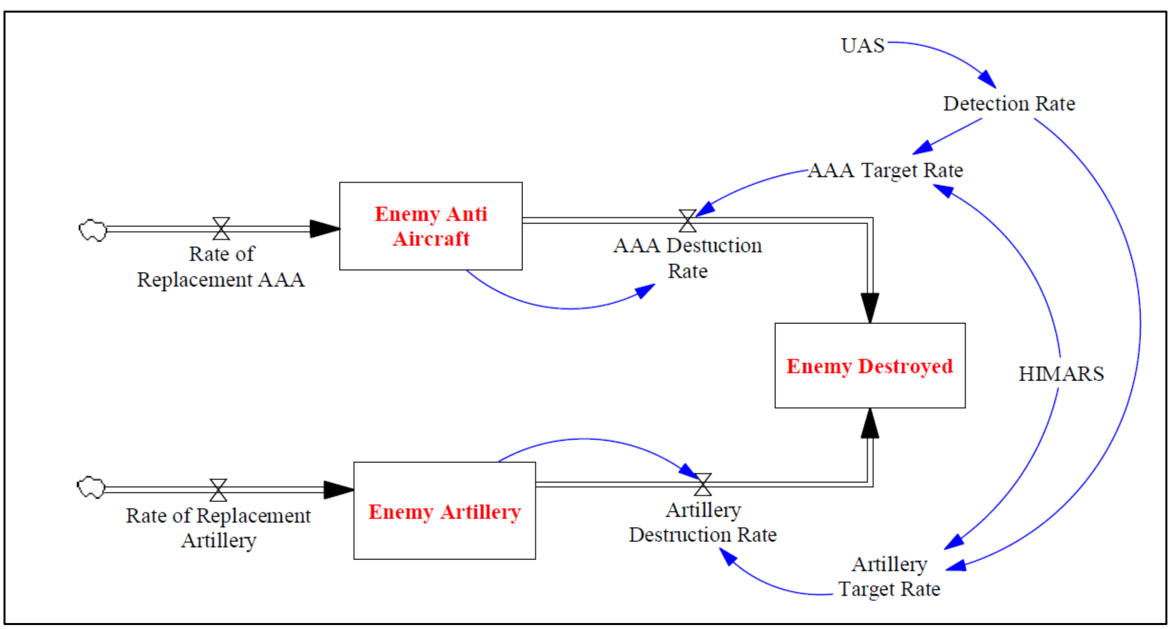


Figure 102: Initial Enemy AAA and Artillery Stock and Flow Diagrams

This stock and flow diagram can be refined further for additional detail. Like the friendly HIMARS, the Enemy Anti-Aircraft and Enemy Artillery should have some variables that influence the rate of replacement to aid in gaming with the ability to set a value for the comparison of technology and policy alternatives. The rate of replacement can be assumed to be directly proportional to the rate of destruction; there is no need to replace assets unless they are destroyed. To provide a means to adjust this rate, and "Ability to Replace" converter is added from 0-100% to either shut off the flow of replacements or have them match the losses 1:1. Additionally, a delay is added for the user to adjust the reaction time to replace. As mentioned previously, these delays can have a significant impact on dynamic behavior and use similar equations for the previous delays.

$$
\begin{aligned}
&Target\ Overflow\ Rate(t) = IF\ THEN\ ELSE(AAA\ Targeting\ Rate \\
&= 0 : AND : Artillery\ Targeting\ Rate\ = 0 : OR : Targets \\
&> \ Max\ Engagements\ , PULSE\ TRAIN(\ Time\ , TIME\ STEP\ , 1\ , \\
&FINAL\ TIME\ ) * INTEGER("\%\ Targets\ Expired\ per\ Hour" \\
&* Targets)/TIME\ STEP\ , 0\ )
\end{aligned}
\tag{14}
$$

The larger and more complex additions come via the expansion of the targeting process (Figure 103). A stock of post-processing targets is generated from UAS ISR operations via the PED, MDTF, and Fusion Cells as identified in the DoDAF operational and system viewpoints. Some percentage of those targets will be beyond the LTIOV due to target backlog and will outflow from the target stock as a Target Overflow Rate. This outflow will become an inflow into Target Overflow that exists a bookkeeping stock to track and measure the aggregated quantity of lost targets. The Target Overflow Rate is a function of Targets, the Targeting Rates, the Max Engagements rate, and a user-defined average Percentage of Targets per Hour (see Figure 103 and (14)).

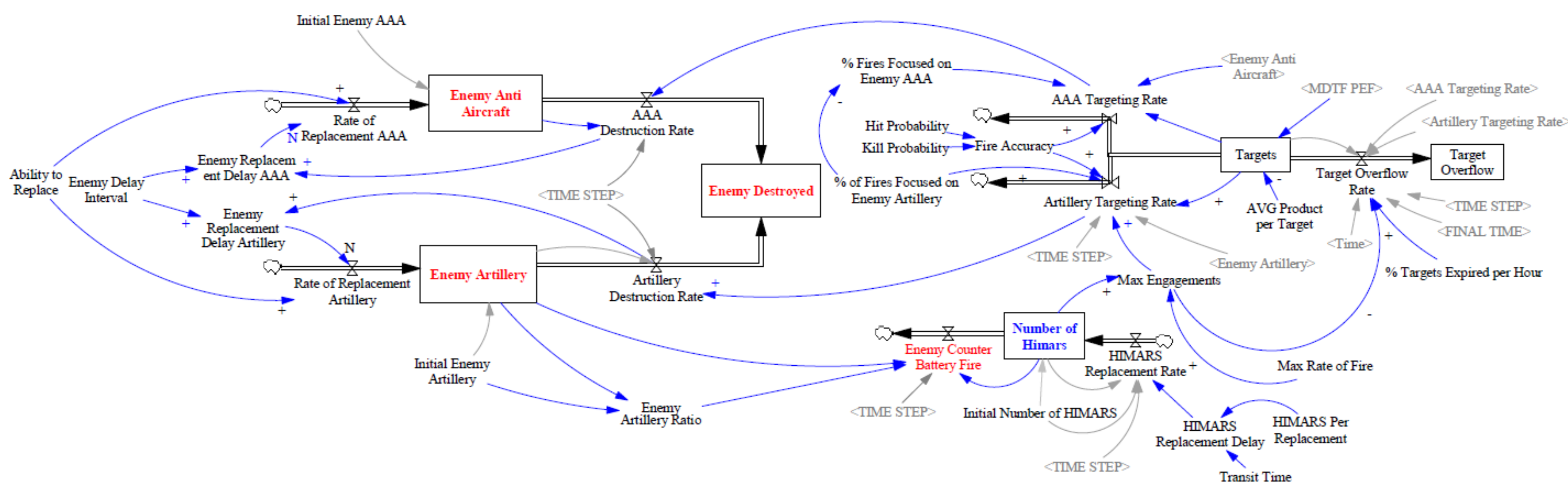


Figure 103 Updated Enemy Subsystem Stock and Flow Diagram

Because, SD works in aggregates, which targets are lost is not specified, so an average rate must be assumed. This expiration rate can be assumed to be due to a variety of factors: target no longer in the area, redundant targets, or late reporting. A throttling variable (Average Products per Target) is added to the Target stock to account for the deluge of fused intelligence products since targets typically require multiple products (fused or directly from the MDTF). This average value is assumed and can be adjusted by the user based on historical data, intuition, or to observe impacts on MOP/MOE.

$$
\begin{aligned}
&AAA\ Targeting\ Rate(t) = \\
&IF\ THEN\ ELSE(\ Enemy\ Anti\ Aircraft > 0{:}AND{:}Targets \\
&\geq 1\ , MIN((RANDOM\ BINOMIAL\ (0, MIN(\ Targets\ , Max\ Engagements\ ), \\
&\qquad Fire\ Accuracy, MIN(\ Targets\ , Max\ Engagements\ ), 0, \\
&\qquad\quad \text{"}\%\ Fires\ Focused\ on\ Enemy\ AAA\text{"}\ , 0\ ) \\
&\qquad\quad /TIME\ STEP), \text{"}\%\ Fires\ Focused\ on\ Enemy\ AAA\text{"} \\
&\qquad\quad * Targets/TIME\ STEP)\ , 0\ )
\end{aligned}
\tag{15}
$$

It is in this enemy stock and flow diagram, that the connection of subsystems begins to take shape. The rates at which Targets are prosecuted are functions of target selection, Fire Accuracy, and the Max Engagements Rate. The Hit Probability and Kill Probability are user selectable values that can be adjusted to desired values. These values can remain fixed for analysis or be varied as targetable metrics to focus more specific efforts. The Max Engagements are tied directly to the Number of HIMARS stock and the Max Rate of Fire for a given weapon system. Again, this can remain fixed or varied for analysis.

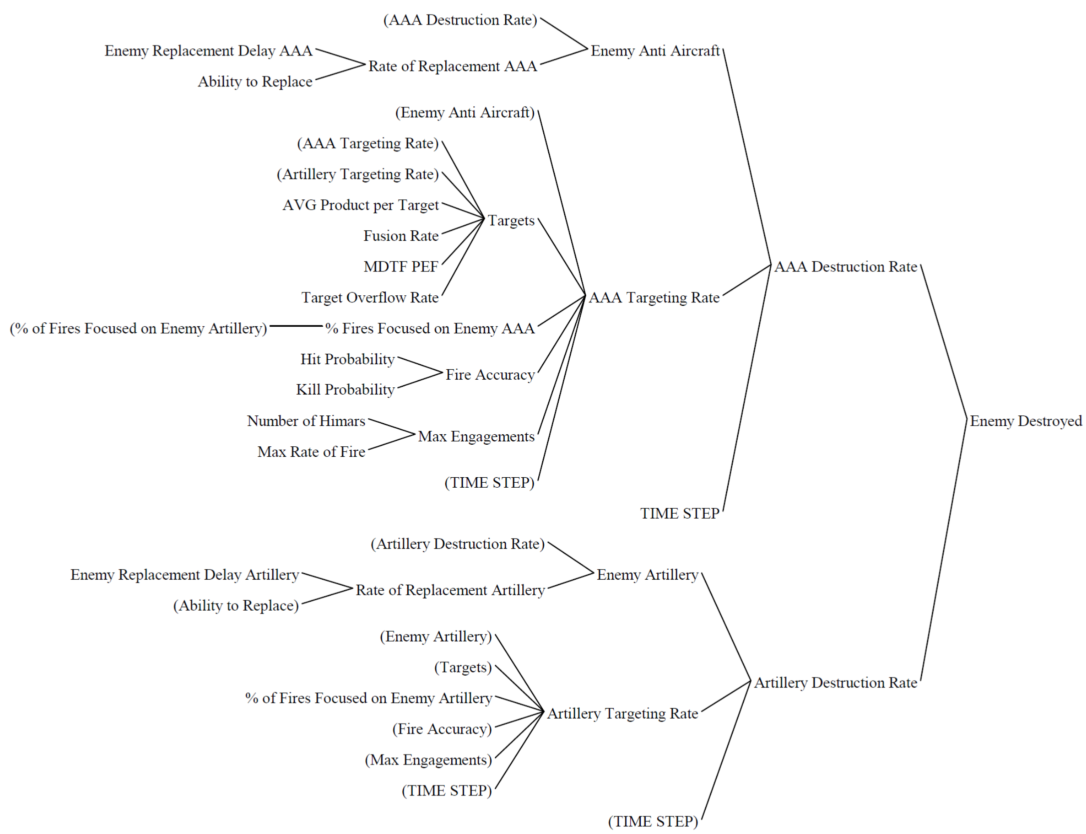


Figure 104 Enemy Destroyed Causal Tree

Lastly, it is via the fire focus ('% of Fires Focused on Enemy Artillery' and '% Of Fires Focused on Enemy AAA') that the decisionmakers and stakeholders can play games and assess policy or the effects on tactics. As mentioned previously, the HIMARS must decide how to focus fires on enemy targets: should 100% of fires be focused on the elimination of Enemy Anti-Aircraft to reduce UAS losses that are vital to targeting, should they focus 100% of fires on the elimination of Enemy Artillery to reduce HIMARS losses, or is it some combination thereof? The equation for the AAA Targeting Rate outflow that ties directly into the AAA Destruction rate is shown in (15). The causal tree in Figure 104 Enemy Destroyed Causal Tree shows the variables the influence this metric.

4.6.3 PED

The critical subsystem that ties all of the other subsystems together is the PED and subsequent fusion of information into intelligence products and, subsequently, targets that can be processed completing the dynamic SoS circle. Using the DoDAF products from Section 4.3.2 and the basic UNCLASSIFIED doctrine [147, 148] and initial PED Stock and Flow Diagram is generated. Each step of the PED Process is a stock representing a delay from the initial inflow (Sensing) to the final outflow/inflow (Dissemination) of the final 'Products' stock (See Figure 105). As was discussed in Section 4.6.1 for the modeling friendly UAS and HIMARS, the PED backlogs represent a pipeline. The rate at which observables collected by ISR platforms is transported between stocks is a function of some average processing rate per person times the number of personnel per stage.

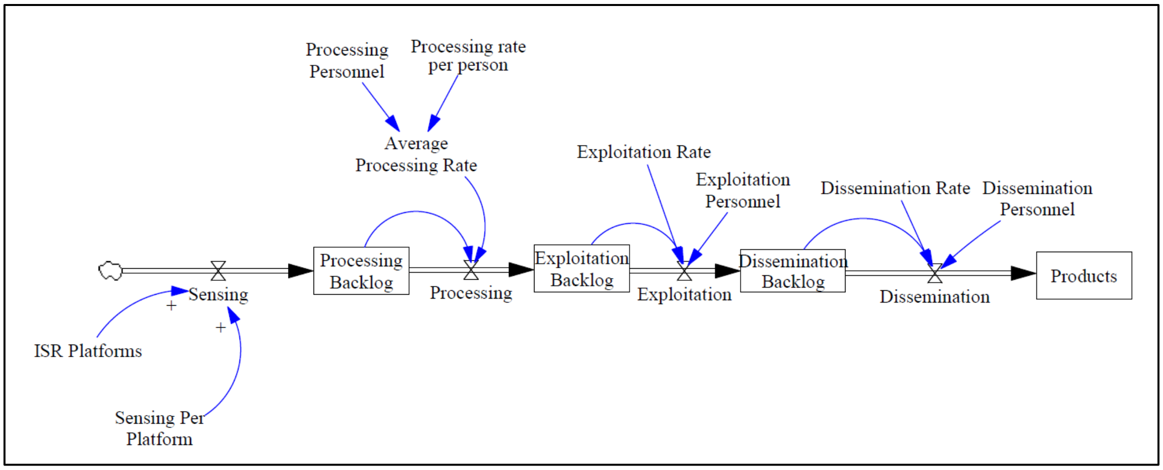


Figure 105 Initial PED Stock and Flow Diagrams.

While this type of PED model could be considered sufficient for an initial holistic assessment of the overall SoS it leaves little in the ways of option for incorporating improvements, specifically in terms of policy, manpower allocation, and technological. Fortunately, some research has been conducted in the area of designing and improving PED, conducted by researchers at Charles River Analytics [122, 123] at the behest of the U.S. Army Intelligence Center of Excellence. These researchers conducted a detailed study of PED processes, conducted numerous site visits, and surveyed SME, key decisionmakers, and stakeholders. From this research, they developed and proposed various SD models to simulate and evaluate PED processes and technology integration. While several models were generated, their “General TPED Systems Model” [123]is the most suitable for this current study.

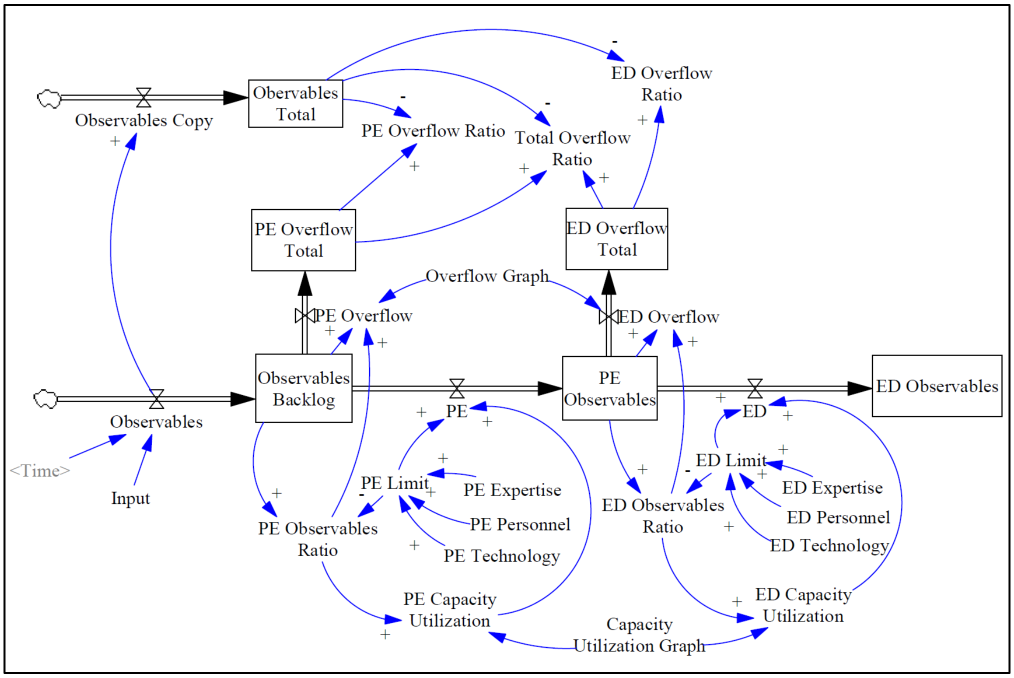


Figure 106: General TPED Systems Model Recreated From [123]

For the purposes of this study, this generalize TPED model provides the right fidelity to inform a holistic, strategic-level analysis for CBA while providing suitable variables that correlate with the initiatives shown in Figure 30: UAS Integration Roadmap Schematic from [11]. In its current construct it provides an ideal starting point to add the MDTF for Research Question 3 and explore structural changes in the executable architecture (see CHAPTER 6). For the sake of brevity, a complete discussion as to how each element of this model was generated, more detailed descriptions can be found in the references [122, 123]. However, key modeling elements will be briefly addressed, as the methods can also be used, as necessary, to refine the larger executable architecture.

In their study, the Charles River Analytics researchers found that while PED is typically delineated as three separate elements, in reality, the initial exploitation is conduction in conjunction with processing, while further exploitation is conducted by senior-level analysts as a final step prior to dissemination. Hence, they consolidated the process into processing-exploitation (PE) and exploitation-dissemination (ED) as the two backlog stocks. The remaining stocks exist to calculate MOPs and MOEs. The rates at which processing occurs (flows) are the important characteristics that can be modified to improve the system and reduce the backlog and subsequent overflows (lost observables due to LTIOV).

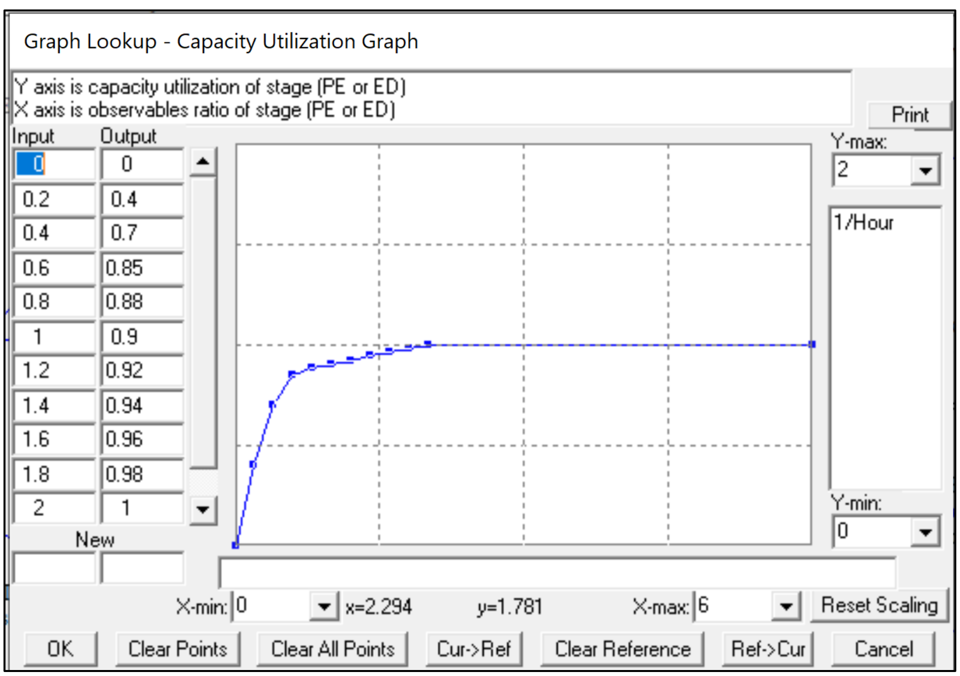


Figure 107: Capacity Utilization Graph recreated from [123]

The PE and ED rates are restricted by PE and ED limits which are dictated by the expertise, personnel, and technology factors. The ratio of the respective backlog to the associated limit determines the capacity utilization for each sector. This utilization is determined by use of an input graph (Capacity Utilization Graph) which is assumed to be the same for both PE and ED. This capacity graph will also be assumed for the MDTF PED once incorporated.

After calculating the observables ratio, the value is entered into the graph on the abscissa and outputs the value on the ordinate. The value on the abscissa represents the observable ratio of the stage. Each Stage must perform work on the observables flow with a limit to the number of observables it can work on per hour. The value on the ordinate represents the percent capacity utilization of stage (PE or ED). It is assumed that a 1:1 ratio of observables to the respective limit constitutes the analysis working at 90%, keeping a 10% capacity in reserve for flexibility and to reduce chronic fatigue. A ratio of 2:1 is the maximum rate that the analysts can process observables for a brief period before reaching burnout; anything over a ratio of 2:1 causes an increase in the respective backlog [123]. The respective capacity limit multiplied by the respective limit will dictate the flow rates (see Figure 106).

The other graph of interest that also influences the backlog stock by controlling an overflow rate. These overflow rates account for observables ratios that are too great for each stage to accommodate, creating overflow. This release valve of sorts accounts for LTIOV. In this graph function (Figure 108). Like the Capacity Utilization Graph, the value on the abscissa represents the observable ratio of the stage. However, in this case the ordinate is limitless, meaning that if there is insufficient analyst capacity to work on them, all the observables can overflow[123]. As with previous subsystems, the observables are aggregated, without differentiation as to which target, sensor type, or specific UAS. There is not means within this system to differentiate priorities of intelligence as would be indicated in the real world, which is a complicated subproblem and beyond the scope of this thesis.

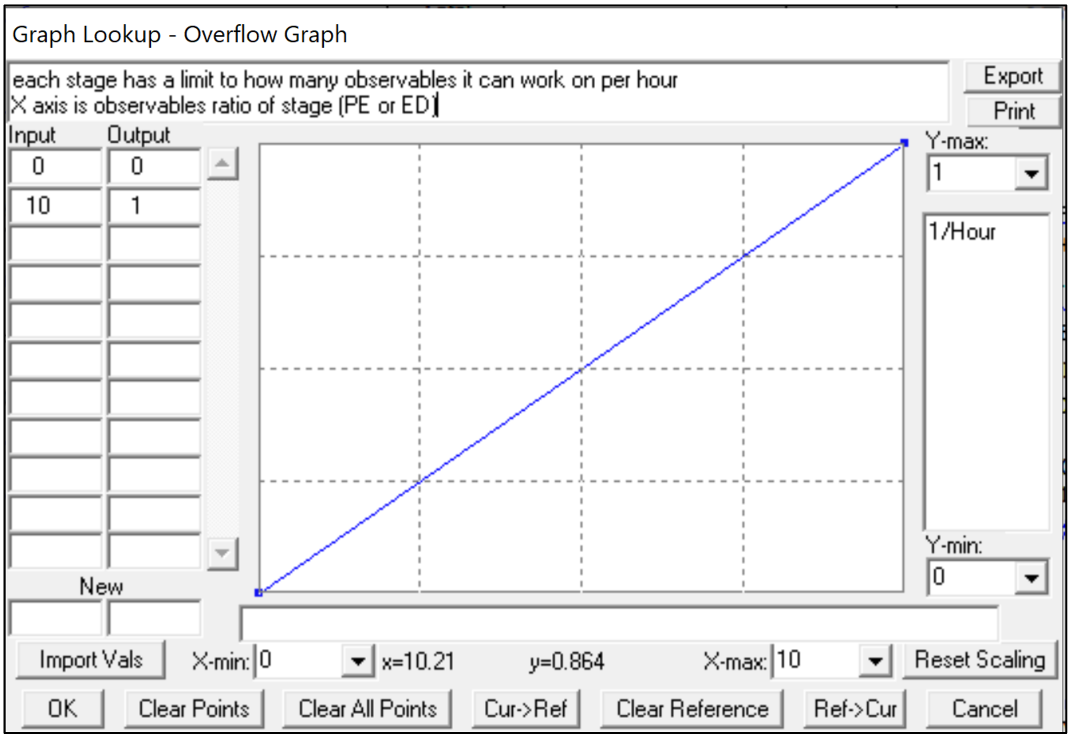


Figure 108: Observables Overflow Graph recreated from [123]

To account for such differentiation, the introduction of the MDTF for prioritized targets is introduced. This alternate system structure is created by generating an additional PED stock and flow diagram with associated connectors (Figure 109). Borrowing from the "General TPED Model", the same Capacity Utilization and Overflow Graphs are used, as are the concepts of observable ratios, overwork, and PE limits. Like the federated PED, the

MDTF PED/Fusion (PEF) has a limit that is dictated by user-adjusted parametric values for expertise, personnel, and technology factor.

Two additional variables are added to connect this stock and flow diagram to the original “General TPED Systems Model”. First is the variable “% Obs Direct to MDTF.” This variable allocates a percentage of the inbound observables from the UAS directly to the MDTF PED rather than through the primary federated PED. This variable accounts for the allocation of priority targets either by some filtering mechanism, or more realistically, by a percentage of direct-support UAS that report only to the MDTF to locate immediate priority targets in the D3A cycle by some designated area. Even without the ability to delineate the aggregated information or to interact with the environment, this will provide some measure to account for these real-world considerations for rapid holistic analysis. The end-user can dictate the percentage of observables allocated or it can be made parametric.

Finally, in a no-growth army with limited human resources, particularly in low density military occupational specialties, personnel will have to be conserved. Namely, to build the PED elements of MDTF, it is assumed personnel will have to be reallocated from the PE Personnel. Hence, the variable “PED Personnel Total” was created along with a “MDTF Allocation” percentage, both of which can be user defined or adjusted (Figure 109).

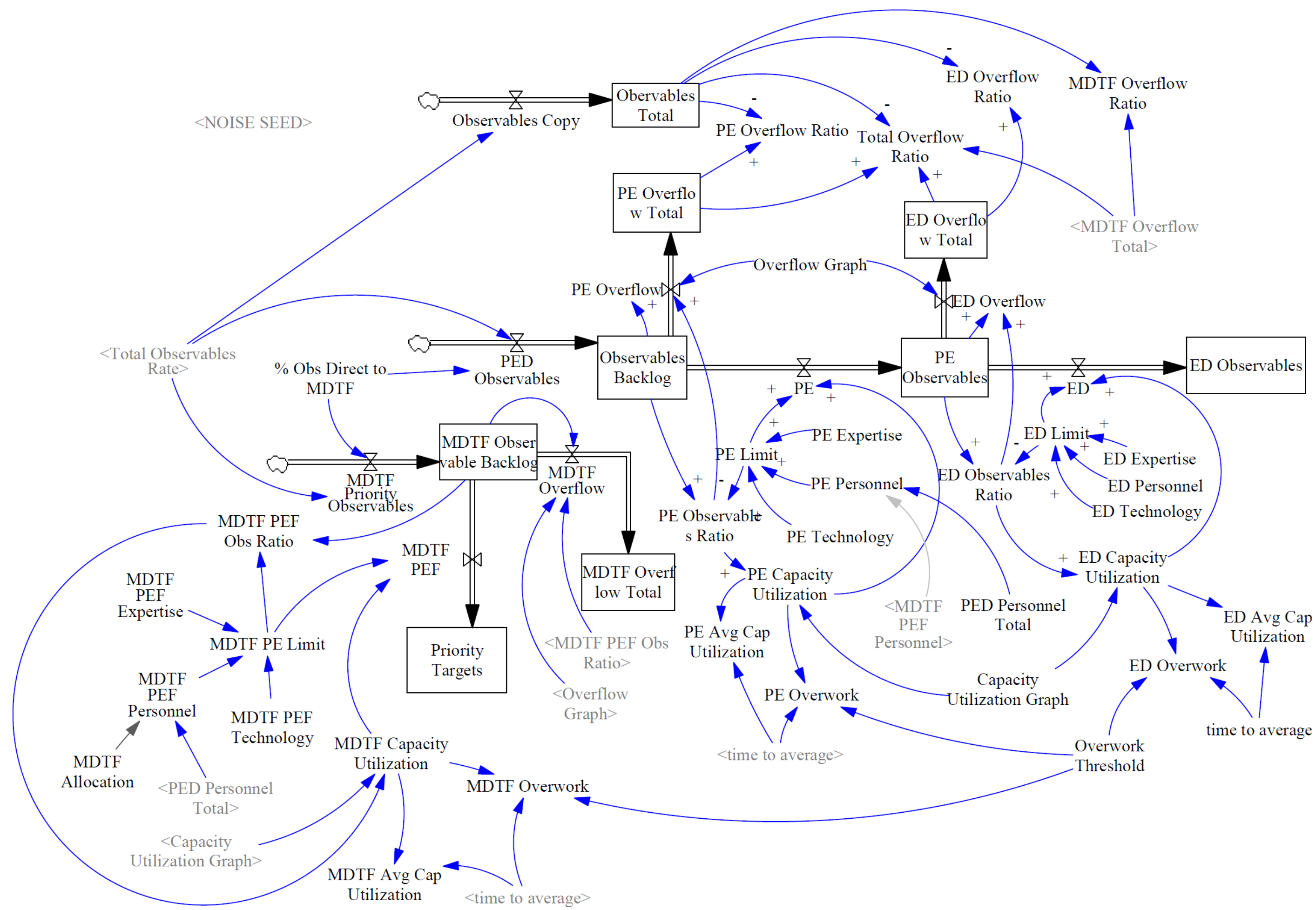


Figure 109 Final PED Specific Sub-Model

4.6.4 Completion of the Overall AISR PED D3A M&S

With the completion of the PED subsystem model, the final model can be created through the integration of some additional stock and flow diagrams and connecters on both sides of the PED. From doctrine, the following the processing and exploitation, the intelligence products are disseminated to the end users via the DISN[147, 149].

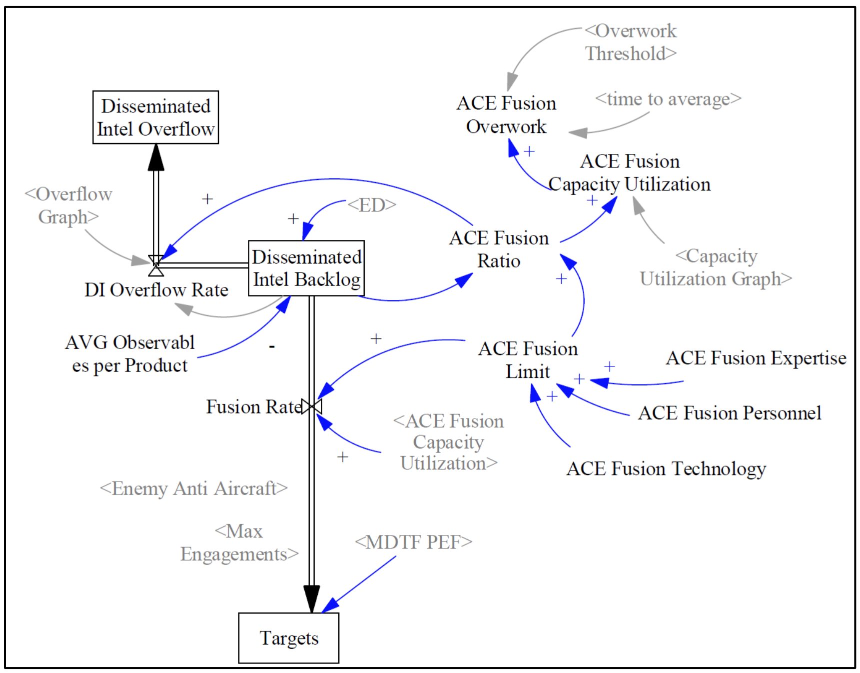


Figure 110: Fusion Stock and Flow Model

At echelons above brigade, the information is fused from various sources at the theater Analysis and Control Element (ACE) prior to being sent to targeting for the purpose of allocating fire support. To replicate this additional delay for the information generated from the federated PED, an additional stock and flow diagram representing the ACE Fusion Figure 110is built in the exact same manner as the PED that connects the PED to the previously created targeting process (Figure 103). Because the MDTF is meant to bypass the traditional fusion for high priority targets, the flow from MDTF PEF feeds directly into the ‘Targets’ stock for immediate execution.

The remaining connection is that of the UAS Overhead to the inflow of PED (Figure 111). This is another potential area where the inclusion of an ABM could provide some benefit, albeit at an added complexity, time, and expense. In lieu of that model, logic is applied. A ‘Target Saturation Ratio’ of targets to UAS is generated to differentiate-rich versus target-sparse environment. For first-order analysis, one could logically infer that a target-rich environment increases the likelihood of detecting enemy assets and convers for target-sparse.

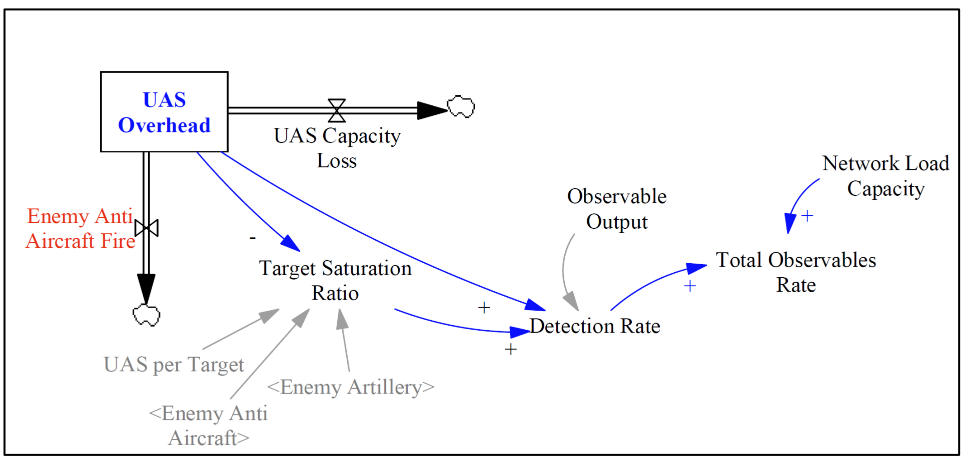


Figure 111: Connectors for Total Observables Rate into PED

The ‘Target Saturation Ratio’ is input to the ‘Observable Output’ Graph (Figure 112) to determine a ‘Detection Rate.’ The ‘Observable Output Graph’ operates on the assumption that the maximum targets a single UAS can track is five per hour regardless of target saturation in target rich environment due to the targeting cycle delay and transit time from one target to another. This graph is easily modified and subject to change based on real data. However, the five targets per hour per UAS is the assumption used in the development of this model and the subsequent experiments.

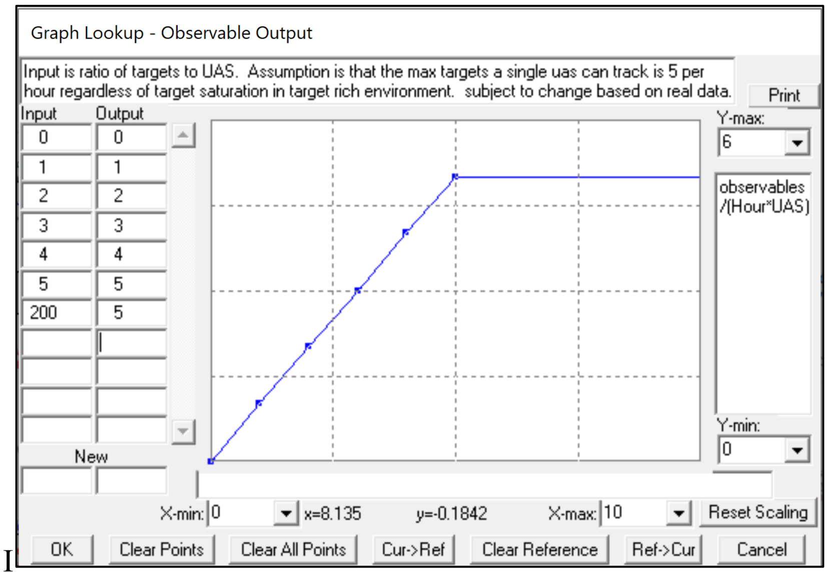


Figure 112: Observable Output Graphical Input

With all the elements combined, the final SD model and simulation environment (Figure 113 and Figure 109) is nearly complete and ready for experiments and analysis. First, however, following Ford [143] and Sterman's [51] model testing and validation methods, the complete model behavior must be tested to ensure the combined SoS is operating in a reasonable manner. This will be demonstrated in the subsequent chapters to baseline the models prior to the conduct of experiments and findings in support of the research questions for the use case to demonstrate the efficacy of the method for developing an executable architecture that was described in this chapter.

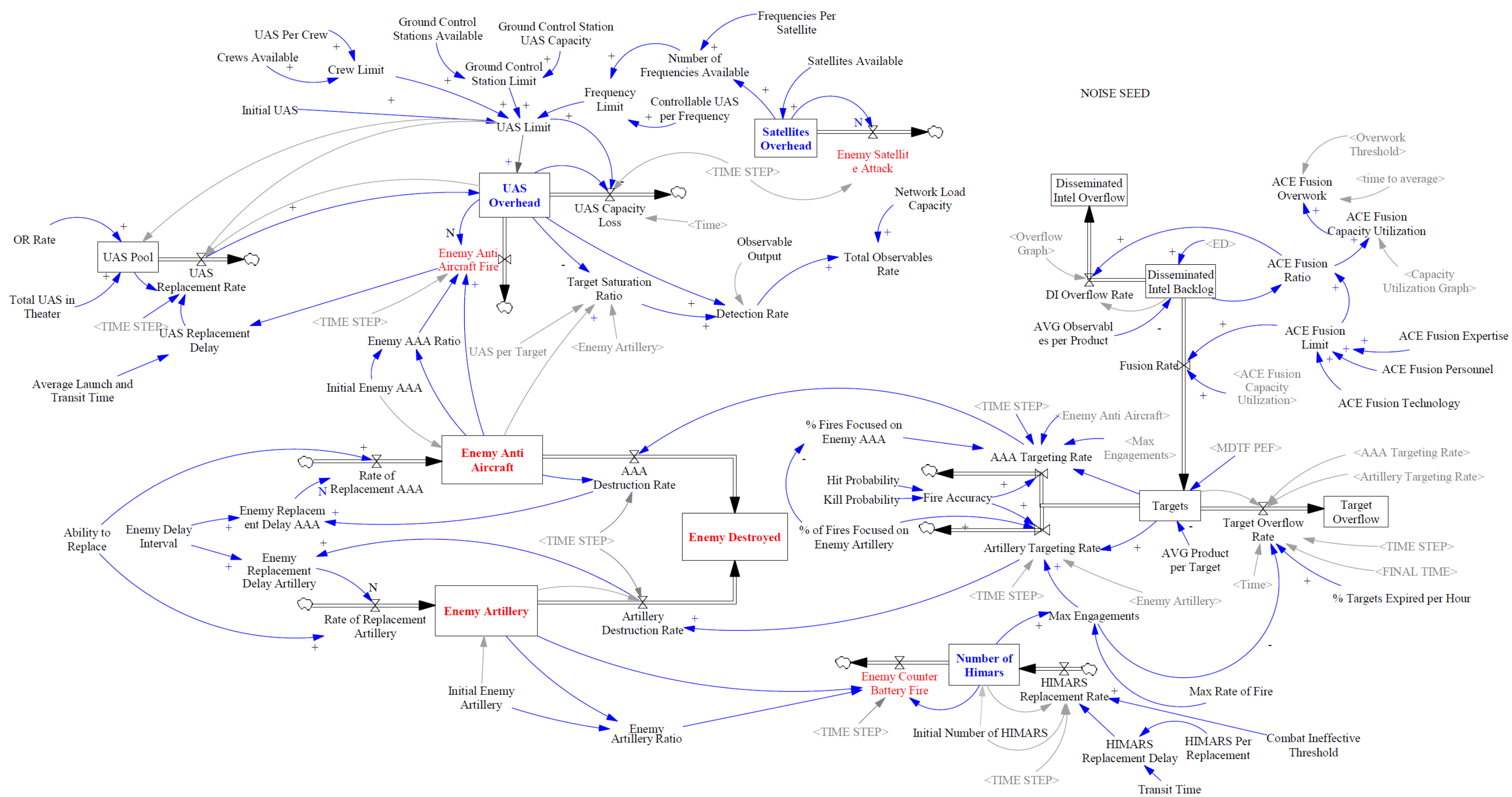


Figure 113: Complete ISR PED D3A Model

4.7 Problem 1 Results and Findings

Problem 1 served as the framework development and apparatus design. Using sample DoDAF models/products, SD was selected and used to develop an executable architecture. This executable architecture was validated using reasonable performance evaluations for each subsection using Sterman's model testing procedures. The resultant M&S environment is determined sufficient to be used to conduct equivalency experiments to demonstrate the potential and efficacy of the executable architecture as a modeling strategic level decision making tool.

CHAPTER 5. PROBLEM 2: POLICY AND STRUCTURE

5.1 Problem 2 Summary

With a complete executable architecture, it can now be used to economically investigate questions and proposed policy, allocation, and technical solutions associated with complex SoS architecture to provide some scientific backing to claims of improvements or needs thereof in the CBA ICD process. The Overall Problem, Hypothesis and summary of Problem 2 are summarized in the figure below for the reader's convenience. The use case will be used to approach and answer both questions 2.1 and 2.2 simultaneously.

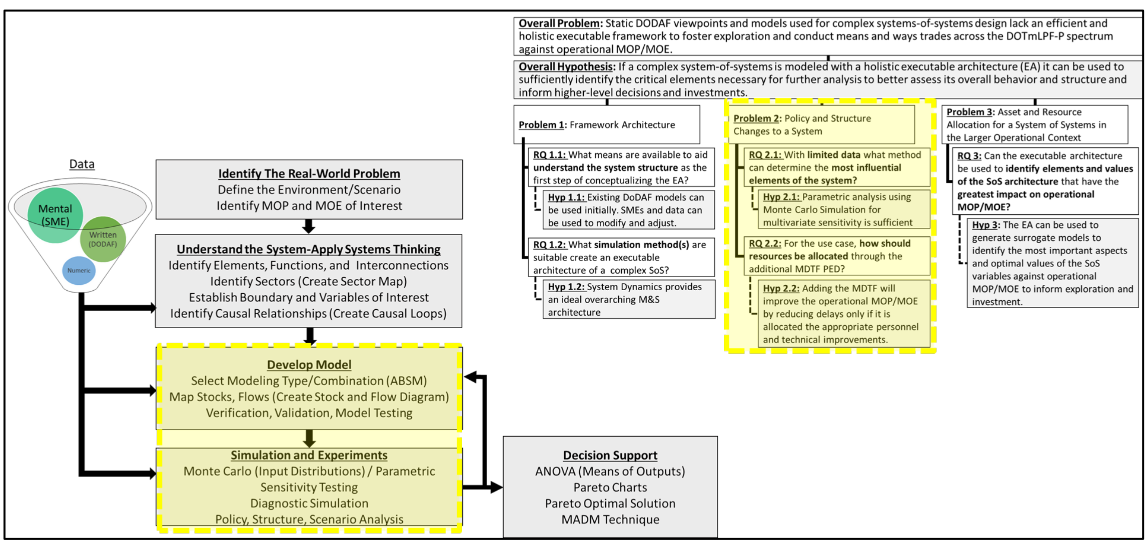


Figure 114: Problem 2 Summary

5.2 Experiment Design

To demonstrate the ability to use the Executable Architecture simulation to foster decisions and play games, the ASIR-PED in a D3A environment. For initial exploration and for demonstrative purposes of rapid comparison with fewer user defined/controlled variables, the PED subsystem-of-systems will be utilized for clarity and brevity. Additionally, since the structural changes in question pertain only to the PED, it is logical to make preliminary estimates based on modification affects to that subsystem only using basic methods before integrating and analysing the large system-of-systems and comparing the effects on operational MOP and MOE. Given that the subsystem is isolated, the MOP/MOE will be those relevant to the improvement of the PED subsystem. The overall MOP/MOE were developed in Section 4.2 and summarized in Table 11 and Table 12. The MOP/MOE germane to PED improvements

Table 19: PED MOP and MOE Recapitulation

PED Centric Measures of Performance
Intelligence Processing Rates
Intelligence Overflow Over Time
PED Centric Measures of Effectiveness
Intelligence Overflow Ratios
Average Intelligence Capacity Utilization
Intelligence Personnel Overwork

5.2.1 Establishing a Baseline

To evaluate the effects of a change in structure, a baseline must be established by which comparisons can be made. For the PED subsystem of the model, outputs from the validate model sans MDTF exists from the research [122, 123]. To test the behavior of their 'General TPED Systems Model', the researchers used an assumed standardized observables input profile to observe the effects of delays on the PED process. The assumed observables profile depicts a 24-hour period during which the first two hours and the last two hours have zero observables, from hours two through twenty-two, the observables increase linearly to a peak of 50 observables per hour at the 12-hour mark and decline at the negative of the slope until the 22-hour mark (Figure 115). Using standardized input functions is a useful way of observing and comparing behavior caused by the system structure with less noise that could mask the sources of behavior [51, 142].

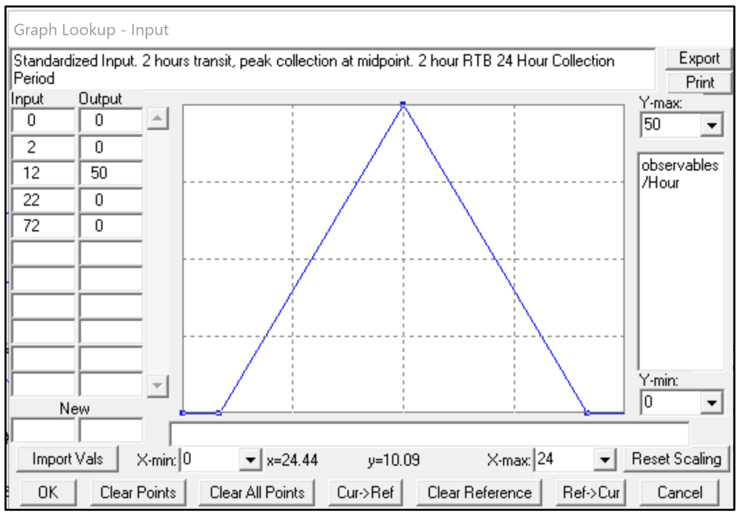


Figure 115: Standardized PED Test Input recreated from [123]

Using PED model depicted in Figure 109 the initial values were set to mimic the behaviors demonstrated by the Charles River model Table 20.

Table 20: Initial Baseline Settings for PED Subsystem Model.

Baseline Run Values							
PE		ED		MDTF		Global Parameters	
PE Expertise	1.5	ED Expertise	3	MDTF Expertise	0	% Obs Direct to MDTF	0
PE Personnel	25	ED Personnel	12	MDTF Personnel	0	PED Personnel Total	25
PE Technology	1	ED Technology	1	MDTF Technology	0	MDTF Allocation	0

For the baseline run there is no MDTF, so the global parameters associated with the MDTF are set to zero, shutting off the flow and personnel to that structure. This action also renders any other settings for the MDTF irrelevant, but for completeness, they are also set to zero. To confirm, the variables were altered to ensure correct correlations and no change in behavior of the model. In the PED Subsystem Model, it is assumed that the number of PED personnel is a fixed number. This is a very relevant constraint for two reasons. First, as mentioned previously in the creation of the model, in a no-growth Army for low density, specialized personnel, a newly formed structure/organization must be built at the expense of another. Second, it eliminates the trivial solution of simply increasing manpower to increase output. Such a solution is limited not only by available manpower, but by workstations, budgets, etc. Because the 'PE Personnel' and 'MDTF Personnel' are functions of 'PED Personnel Total' and 'MDTF Allocation' they are greyed out to indicate they are not user-adjustable but are include in the table for completeness.

For the baseline, the technology factors are all set to a value of one. It is assumed that for the initial assessment, no advance technology exists to improve the rate at which personnel can process, exploit, or disseminate the observables or to act as a force multiplier against the fixed number of personnel. The bulk of the personnel reside at the first stage of PE. However, the experience level is lower at this stage and limits the amount of observable feeds that the personnel can process per hour per person. The ED stage is assumed to be comprise of more seasoned and skilled personnel that can refine and complete the products produced from the raw observables at the processing stage. Because this final stage can exploit and disseminating at a higher rate and contains more senior personnel, it has a fewer number of personnel than the PE stage.

The graphs in Figure 116 depict the throughput for each stage of the PED process through the federated PED given the inputs annotated in Table 20. The center graph shows the input observables per hour as was depicted in Figure 115. The blue line indicates the processing rate of the PE flow. As expected, it is shifted in time due to the delay with its slope running generally parallel if only slightly more shallow to the input flow, indicating it the PE stage is capable of processing the information at the rate it is received.

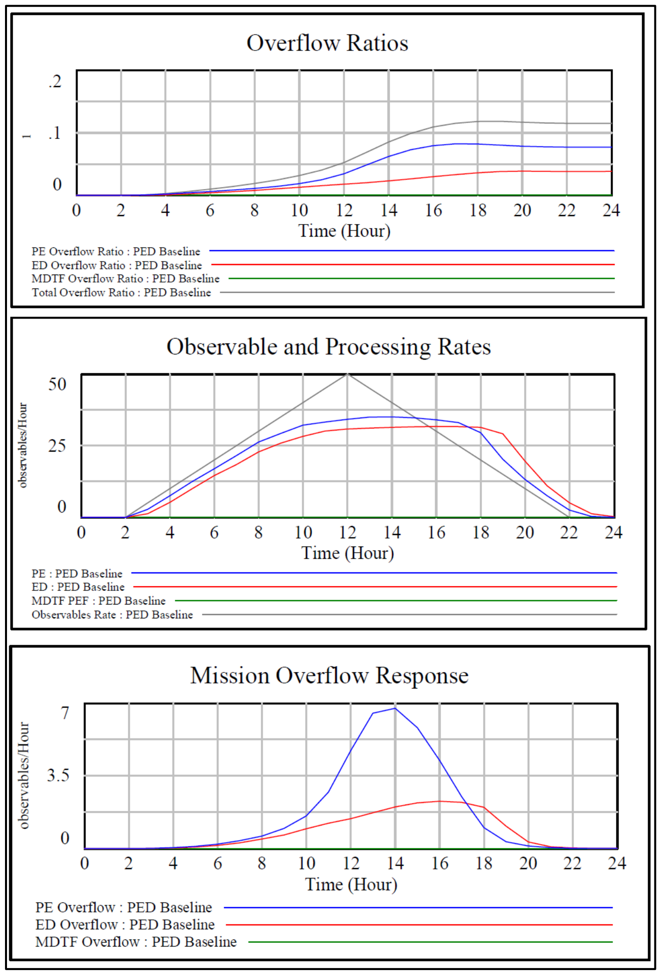


Figure 116: PED Baseline Processing Rates and Overflow

At approximately the 9-hour mark, the PE flow rate begins to flatten out below the rate of incoming observables indicating that the PE is reaching its PE Limit and operating above its threshold. The PE capacity utilization will ultimately influence the PE flow rate and the ‘Observables Backlog’ (see Figure 117), capping the throughput , increasing the overflow. These behaviors are reflected in the flattening of the curve in the center graph along with the increased overflow response in the bottom graph representing unprocessed information. The top graph indicates the increasing overflow ratio representing the percentage of information that goes unprocessed. Note that this ratio increases as the PE attempts to process the backlog after the peak of the input flow eventually leveling off at a ratio of around 0.08. The peak mission overflow response (bottom graph Figure 116) is delayed approximately two hours from the peak input and the primary bottleneck.

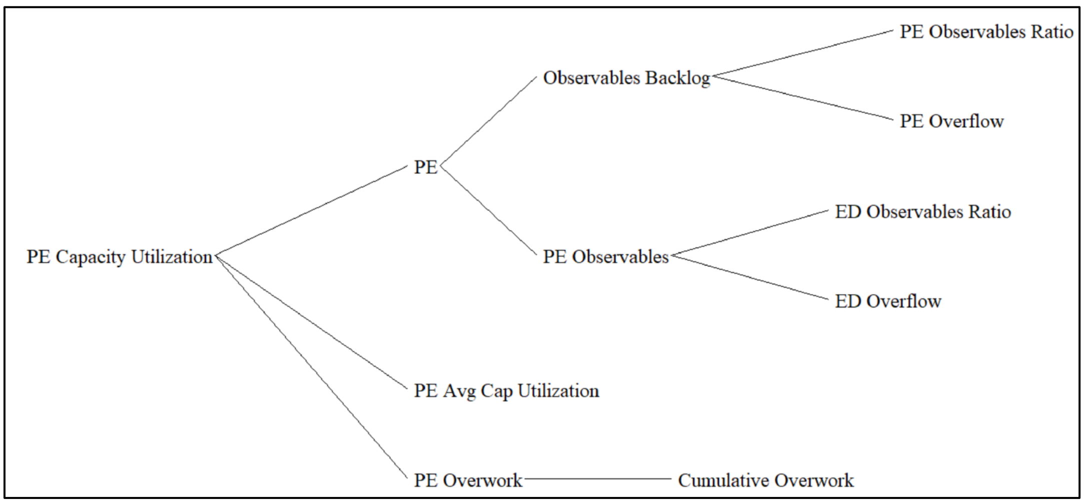


Figure 117: PE Capacity Utilization Uses Tree

The PE Overflow is a direct reflection of exceeding capacity of 100%. In addition, however, a general heuristic is that 85% capacity utilization is the ideal maximum for sustained operations to prevent fatigue and provide flexibility (this value can be changed by the user but is kept at 85% for the entirety of this thesis). The graphs in Figure 118 depict the Capacity Utilization for both PE and ED in the top and middle graphs. The middle graph ‘Average Capacity Utilization’ simply applies a smoothing factor to account for variability noise. Since the input is already a smooth function, the graph is identical to the ‘Capacity Utilization’ graph above it. Depending on the severity of noise, this smoothing factor can be adjusted to provide clearer insight into the workload on the PED force unadulterated by random spikes. For the baseline, this factor is set to a value of one. The bottom graph in Figure 118 more accurately captures the ‘Overwork’ above the ideal ‘Overwork Threshold’ of 85%.

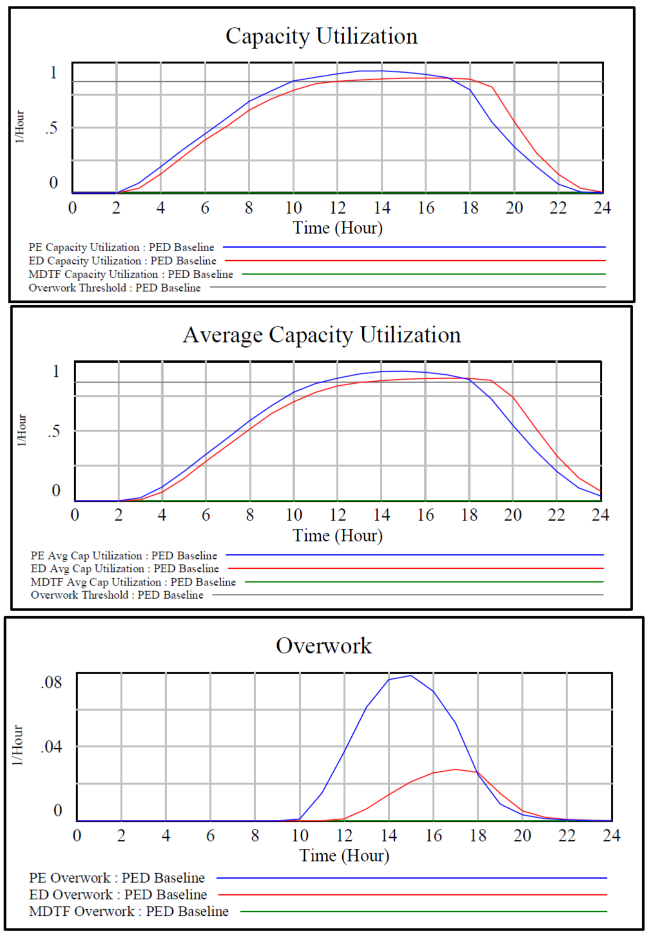


Figure 118: PED Baseline Capacity Utilization and Overwork

Finally, to establish the baseline, it is imperative to discuss the standard expected behavior of the ED indicated by the red lines on all the graphs in Figure 116 and Figure

118. First, due to the delay associated with the PE process, the ED functions are further shifted later in time, initially by about 10 minutes as production is able to keep up with input, until the delay is exacerbated by the PE working at full capacity, at which point the ED is also working above the "Overwork Threshold.' This delay is indicated in bottom graph of Figure 118 with approximately a two hour shift in peak overwork at (approx. 17.5 hours) correlating to the peak processing rates and overflow. Note that the 'ED Overflow' is less than the 'PE Overflow' despite having fewer personnel (Figure 116 bottom). This is due to two factors: first, despite having fewer personnel and the same technology factor, the ED stage has higher expertise; secondly, the ED has a lower amount of observable products to analyze due to the PE Overflow at its peak at 14 hours (Figure 116 bottom). As with previous graphs, the peak ED Overflow is shifted by two hours (at approx.16 hours) due to the delays from the input through the PE. Note that the overflow peak proceeds the overwork peak as the ED stage attempts to "catch up" until the backlog stock is worked off.

5.2.2 Adding and Testing Structural Changes

5.2.2.1 Setup

Having established the baseline behavior for the MOP/MOE of interest, the additional structure can be test and behaviors observed for predictable variations. By varying the '% Obs Direct to MDTF' and 'MDTF Allocation' (see Table 21).

Table 21: PED Subsystem Model MDTF Test 1: Equal Allocations.

Equal Allocation of Obs and PAX to MDTF Run Values							
PE		**ED**		**MDTF**		**Global Parameters**	
PE Expertise	1.5	ED Expertise	3	MDTF Expertise	1.5	% Obs Direct to MDTF	0.2
PE Personnel	20	ED Personnel	12	MDTF Personnel	5	PED Personnel Total	25
PE Technology	1	ED Technology	1	MDTF Technology	1	MDTF Allocation	0.2

By maintaining the same level of technology and expertise at the MDTF as the PE, while allocating equivalent amounts of observable flow and personnel, we expect to see some predictable results that ensure proper function and confirm insight on how such an allocation affects the system as a whole. The associated graphs from this variation are shown in the figure below.

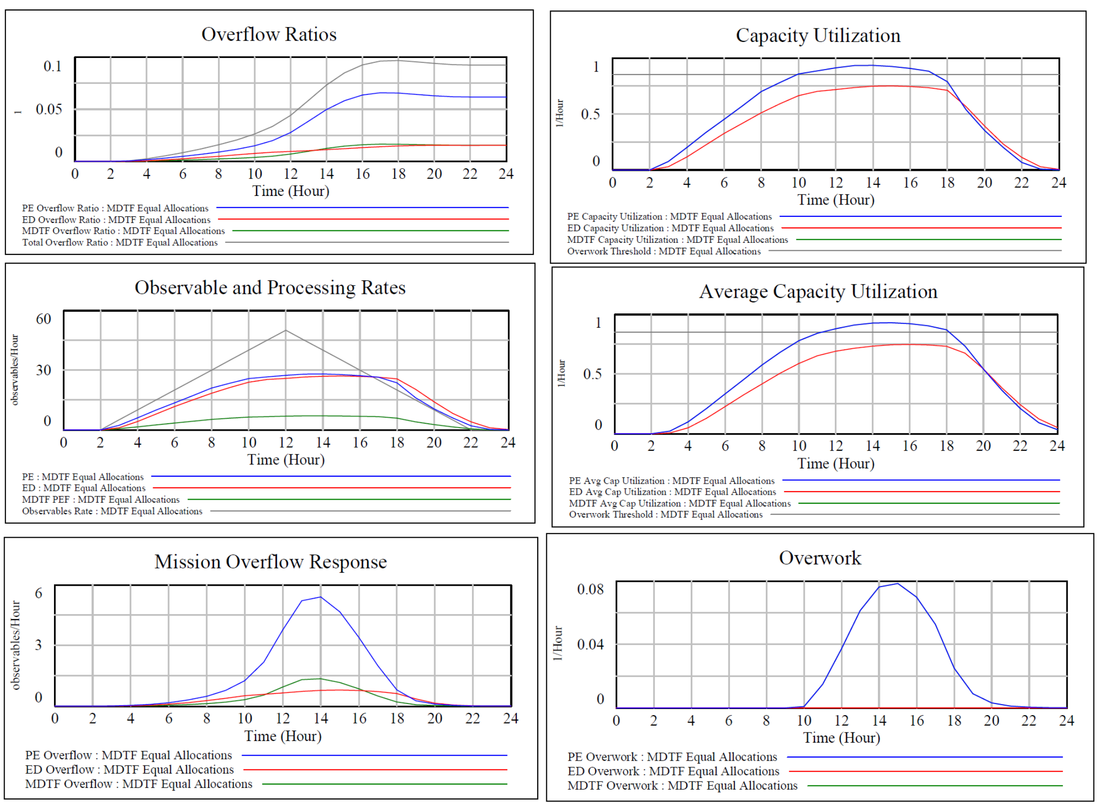


Figure 119: MDTF Equal Allocation Test Responses

5.2.2.2 Observations of Structural Change on Flow Rates

From the graphs some obvious additions are observed: namely the addition of the MDTF (green) on all the graphs which are easily observed in the first column of graphs. In the second column of graphs, the MDTF (green line) is obscured by the PE (red line) as their technology, experience, and observables to personnel ratios are identical. With a quick glance, it can also be noted that when compared to Figure 116 and Figure 118 that while the PE Capacity Utilization (blue line) as remained unchanged (due to the same observables to personnel ratio) the ED Capacity Utilization (red line) has dropped below the 85% Overwork Threshold in the second column graphs in Figure 119. This is further evidenced by the absence of ED Overwork in the bottom graph. For the convenience of the reader, subsequent individual graphs will be used to explore more faceted changes to the outputs that should be noted.

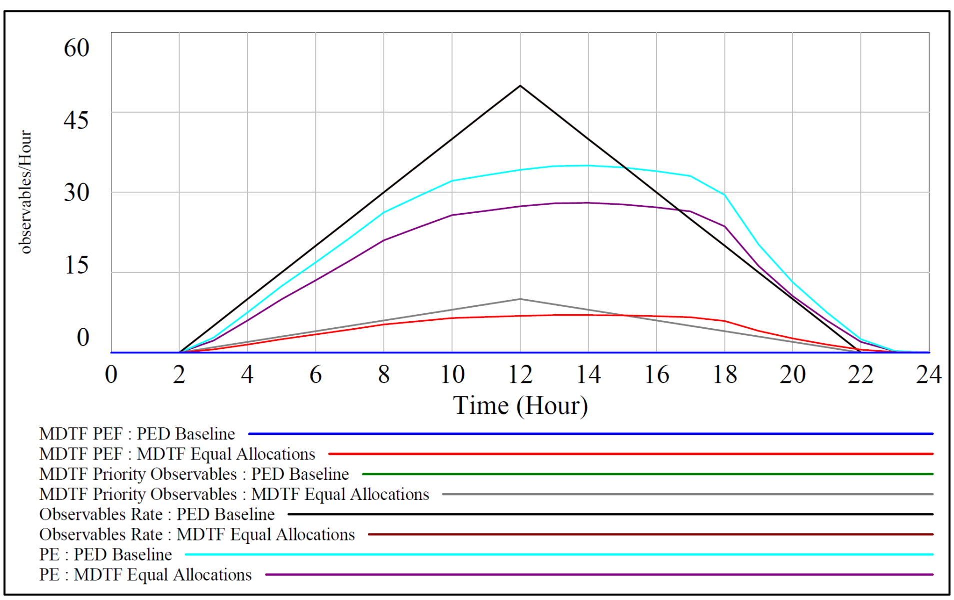


Figure 120: Observable and Processing Rates Baseline to Equal Distribution MDTF

The graph in Figure 120 overlays the Observables and Processing Rates from the PED Baseline run and the MDTF Equal Allocations run. The black triangle represents the total observable flow (see Figure 115) into the PED System with the smaller grey triangle representing the 20% allocation of those observables directly to the MDTF. The turquoise line represents the baseline PE Observable Rate discussed previously. The purple line directly below it represents the PE Observables Rate for the federated PED which has been reduced by 20% as expected due to the reallocation of personnel and observables to the MDTF. As a result, the MDTF PEF Rate increases from zero in the baseline (blue) to a peak of approximately 7 observables/hour at the 14-hour mark as the baseline did.

Table 22: Processing Rates Baseline to Equal Distribution MDTF Tabulated Data

Time (Hour)	Selected	MDTF PEF		PE	
0	Variables	0	0	0	0
1	Runs:	0	0	0	0
2	MDTF Equal	0	0	0	0
3	Allocations	0.558713	0	2.23485	2.79357
4	PED Baseline	1.50287	0	6.01147	7.51434
5		2.48686	0	9.94743	12.4343
6		3.38314	0	13.5326	16.9157
7		4.29615	0	17.1846	21.4808
8		5.24594	0	20.9838	26.2297
9		5.85337	0	23.4135	29.2669
10		6.42714	0	25.7086	32.1357
11		6.6381	0	26.5524	33.1905
12		6.84057	0	27.3623	34.2029
13		6.98782	0	27.9513	34.9391
14		7.00572	0	28.0229	35.0286
15		6.93344	0	27.7338	34.6672
16		6.79708	0	27.1883	33.9854
17		6.6106	0	26.4424	33.053
18		5.90324	0	23.613	29.5162
19		4.0422	0	16.1688	20.211
20		2.63696	0	10.5478	13.1848
21		1.50535	0	6.02138	7.52673
22		0.498233	0	1.99293	2.49117
23		0.0587266	0	0.234906	0.293633
24		0.00693231	0	0.0277293	0.0346616

Comparing MDTF PEF rate to the MDTF Priority Observable triangular input shows the same offset depicted in the baseline. For the second run the sum of the MDTF

PEF rate and the PE processing rate match the PE rate for the baseline run. This is readily apparent by inspection of Figure 120, but can be confirmed empirically via the tabulated date shown in Table 22.

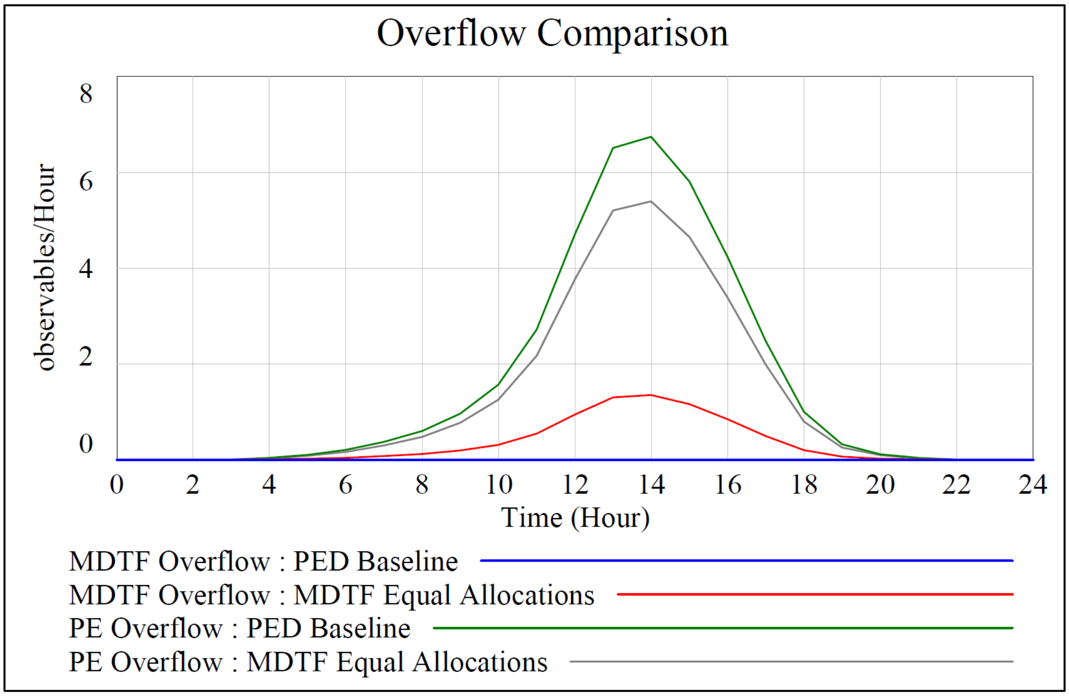


Figure 121: Overflow Rates Baseline to Equal Distribution MDTF

As with the processing rates, the addition of the MDTF with equal load and personnel percentages allocated from the federated PED results in an associated decrease in the PE Overflow (grey) and an increase in MDTF Overflow (red), the sum of which is equal to the MDTF Overflow rate in the baseline (Figure 121 and Table 23).

Table 23: Overflow Rates Baseline to Equal Distribution MDTF Tabulated Data

Time (Hour)	Selected	MDTF Overflow		PE Overflow	
0	Variables	0	0	0	0
1	Runs:	0	0	0	0
2	MDTF Equal	0	0	0	0
3	Allocations	0.00104053	0	0.00416214	0.00520267
4	PED Baseline	0.00752871	0	0.0301148	0.0376436
5		0.0206149	0	0.0824595	0.103074
6		0.041087	0	0.164348	0.205435
7		0.0745196	0	0.298078	0.372598
8		0.119784	0	0.479134	0.598918
9		0.192989	0	0.771958	0.964947
10		0.313326	0	1.2533	1.56663
11		0.542891	0	2.17156	2.71446
12		0.942087	0	3.76835	4.71043
13		1.30105	0	5.20421	6.50526
14		1.34862	0	5.3945	6.74312
15		1.16175	0	4.64701	5.80877
16		0.84712	0	3.38848	4.2356
17		0.497103	0	1.98841	2.48552
18		0.199794	0	0.799175	0.998969
19		0.0642287	0	0.256915	0.321144
20		0.0231785	0	0.0927138	0.115892
21		0.00755355	0	0.0302142	0.0377677
22		0.000827455	0	0.00330982	0.00413727
23		1.14961e-05	0	4.59842e-05	5.74803e-05
24		1.6019e-07	0	6.4076e-07	8.0095e-07

At first glance, the results of Figure 120 and Figure 121 appear trivial. However, they provide valuable insight into the accuracy of the model and to the effects of the additional structure. First, the graphs and their associated values indicated that the structural changes in the model are made correctly and behaving as expected for an equal allocation of personnel and manpower. Second, the change in structure has seemingly no change in the processing flow rates and the overflow rates. Superficially, this observation would indicate the same amount of information was processed and/or lost, leading one to believe that the addition of the MDTF stage and change in structure provides no discernible benefit to ability to accomplish the MOP/MOE. The assertions are partly correct if focusing only on the processing and exploitation stages. However, the benefits of adding the MDTF, (even if only reallocated from the federated PED PE stage and redirecting a proportional rate of observable flow) are demonstrated in the second order effects on ED.

The additional routing structure through the MDTF reduces the amount of observable flow into the PE Backlog stock and, as a result, reduces the delay of processed observables to the ED Backlog. Without altering the number, technology, or experience variables of the ED stage, the reduced backlog results in reduced ED Overflow (Figure 122), ED Capacity Utilization, and the elimination of ED Overwork (see Figure 126).

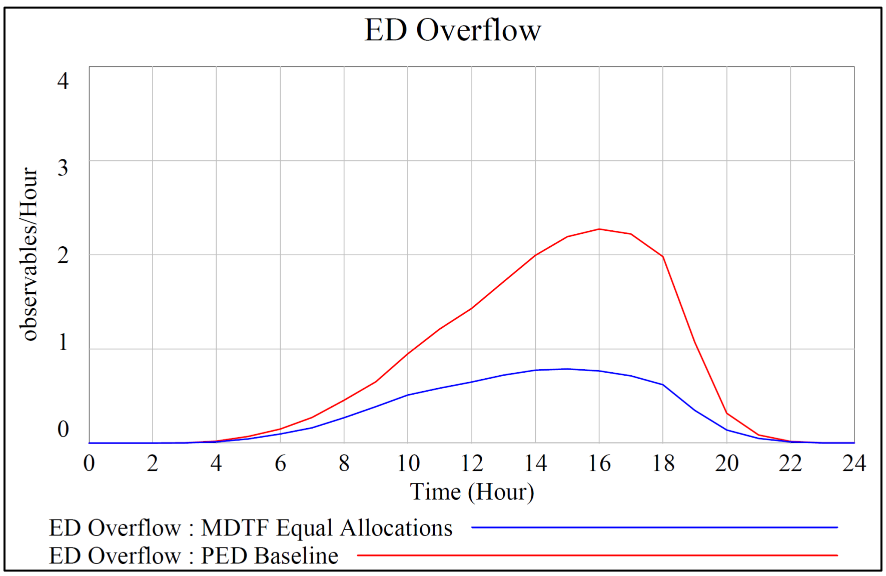


Figure 122: ED Overflow Rate Baseline to Equal Distribution MDTF

From Figure 122 it can be observed that the ED Overflow rate is reduced by over a 50% through the duration. Because the flow is a rate, the integration of that flow overtime, represented by the ED Overflow Total stock (Figure 123). From this graph we can see that over the 24-hour duration the addition of the MDTF resulted in a reduction of approximately 11 total observables lost due to backlog and exceeding LTIOV through inspection of the baseline (red) to the MDTF run (blue).

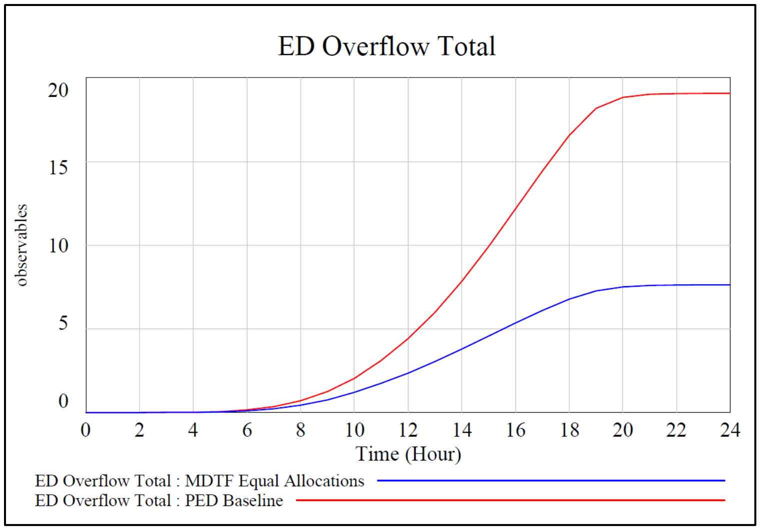


Figure 123: ED Overflow Total Baseline to Equal Distribution MDTF

Because of the decrease in lost observables there must logically be an increase in the total amount of observables disseminated in the final stage. However, looking at the ED Observables alone, it would appear as if there are fewer products being disseminated to targeting (see Figure 124), though this is not actually the case.

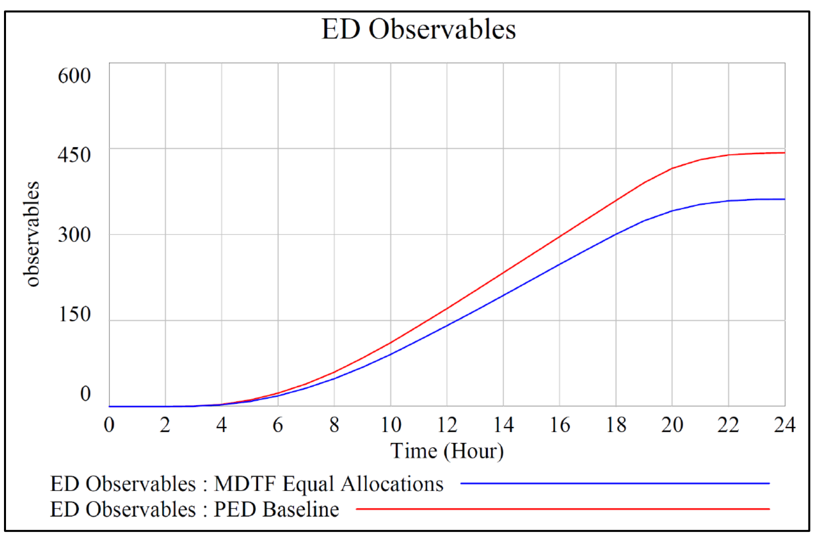


Figure 124: ED Observables Total Baseline to Equal Distribution MDTF

When analyzing a change in structure, it is imperative to account for the alternate sources of information flow. In this case, the MDTF PEF flow directly feeds into the 'Priority Targets' stock (see Figure 109). The ED Observables stock, on the other hand will also feed into the 'Targets' stock after Fusion at the Fusion Rate (see Figure 113). The total processed observables, therefore, is the sum of the MDTF PEF and ED Flows (see Figure 125)

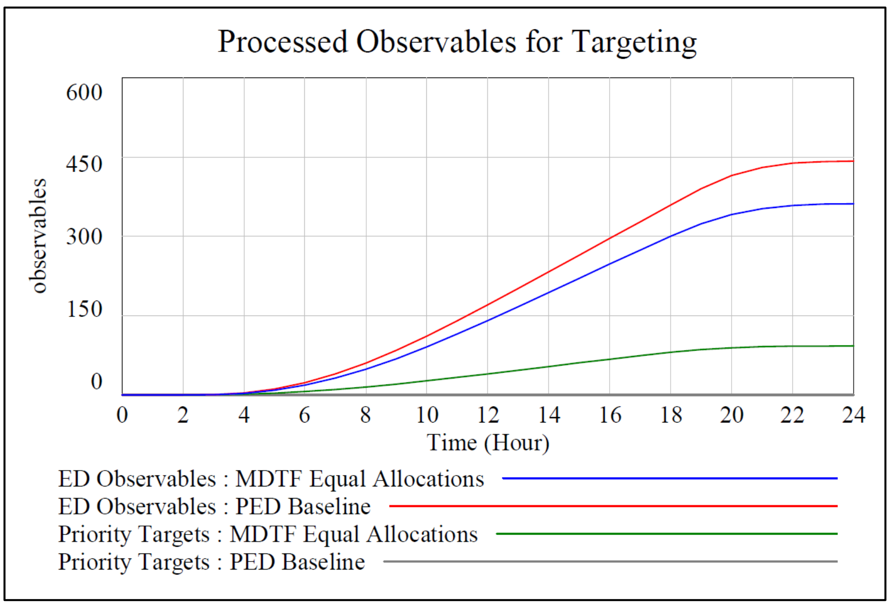


Figure 125: Processed Observables Total Baseline to Equal Distribution MDTF

From the baseline run, the processed observables for targeting only came from the ED Observables stock (red line). With the inclusion of the MDTF, the processed observables come from both the ED (blue) and MDTF (green). The sum of the ED and the MDTF for the new structure totals 453 versus the ED alone prior to the inclusion of the MDTF which totaled only 442 over the course of the 24-hour period. This difference accounts for the additional 11 observables backlog the ED was able to process due to reduce workload. The final number values, however, serve only as estimates based on the assumed inputs and rates. What matters most, is the overall behavioral trends observed, which is an increase in productivity from the alternate routing through the MDTF—even though the total personnel, skill and technology factors remained constant.

Additionally, the alternate routing through the MDTF, and subsequent reduced flow through the ED Capacity Utilization below the 85% threshold for the duration of the operation and eliminated Overwork (see Figure 126) The benefit of this is the ability to sustain long-term operations without fatigue of critical ED personnel.

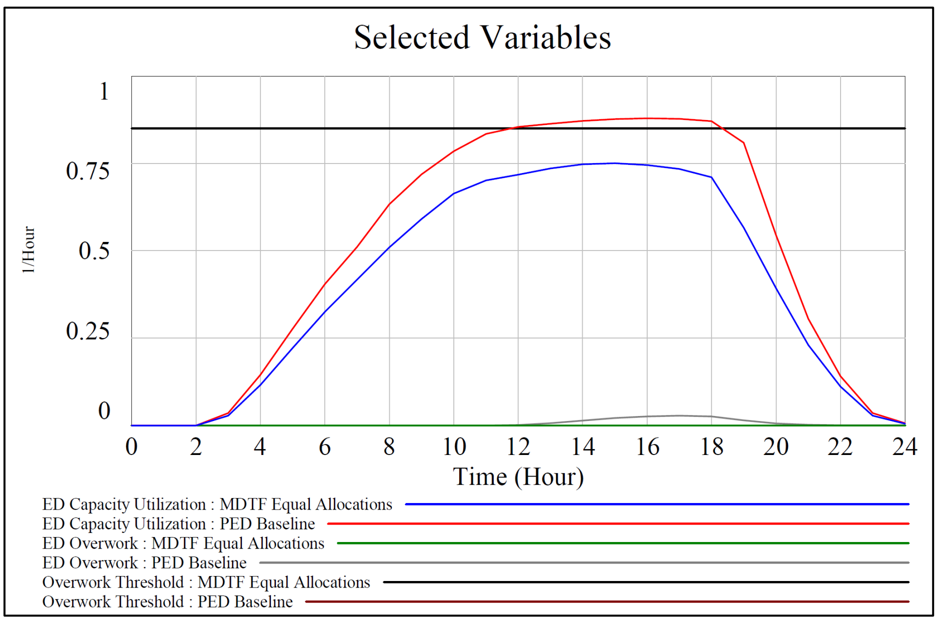


Figure 126 Capacity Utilization and Overwork Baseline to Equal Distribution MDTF

5.2.2.3 <u>Summary</u>

In summary, the previous analysis has demonstrated how additional structure was added to the system and how it was tested for accuracy given the baseline and assumed input flow to the overall system. Additionally, just this simple alteration of the structure through the MDTF generated a net effect that not only reduced stress on the workforce, but an increased number of targetable information distributed to targeting sooner due to reduced ED overflow and reduced number of delays.

5.3 Determining Sensitivity/Most Influential Factors.

5.3.1 Manual Analysis

The simple experiment in the previous section demonstrates a marginal benefit of the MDTF structure to the overall behavior of the subsystem with respect to the PED-specific MOP/MOE, it still leaves many questions unanswered. Having introduced a standardized input and understanding baseline behaviors, experiments can be conducted to test proposals against desirable behavior. Further experimentation can be conducted with the model via a function of Vensim called SyntheSim [142] (for synthesis of simulations) where model structure and simulation behavior can be synthesized and instantly updated and superimposed on graphs as the model constants and lookups are varied (see Figure 127). This function will provide decisionmakers a general understanding of the effects of structure and policy variations.

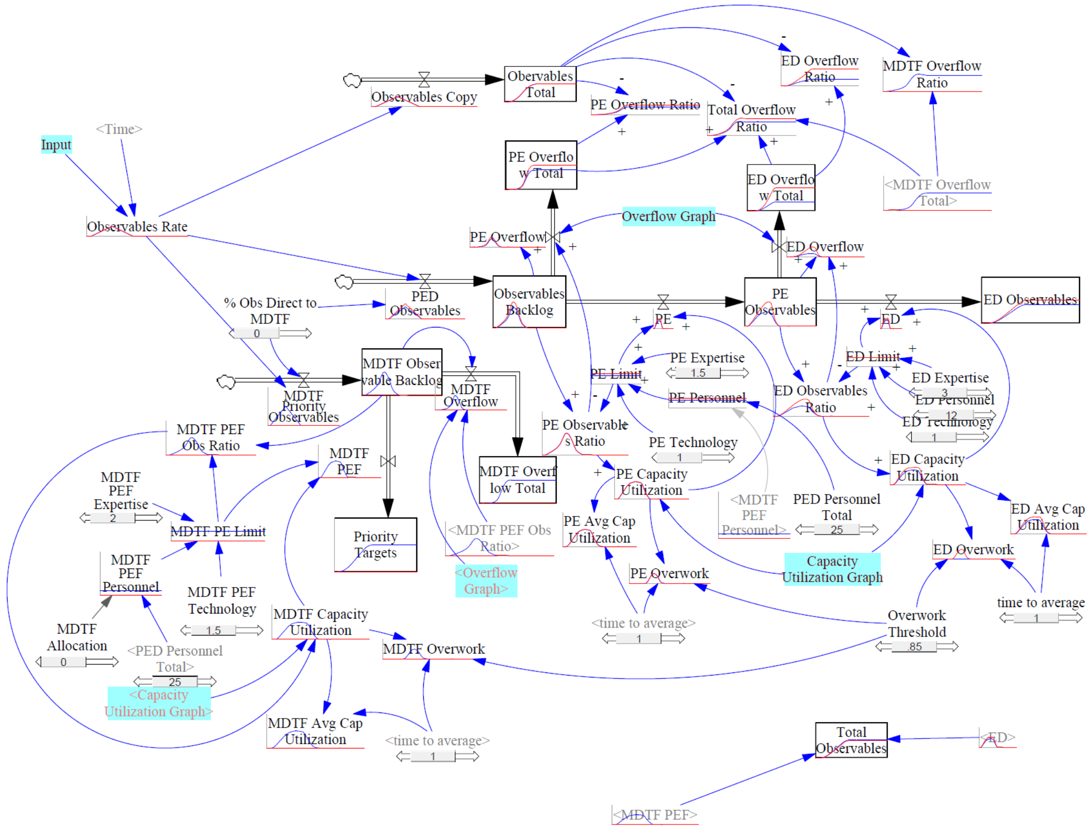


Figure 127 PED Subsystem Model in SyntheSim Model

Simple variation can be made on the previous model to determine policy and allocation decisions. Suppose given the small size but vital importance of the MDTF PED and Fusion to the D3A cycle, decisionmakers deem it necessary to maintain the allocation of personnel and observables but increase the average expertise-level and processing technology at the MDTF.

Table 24: PED Subsystem Model Ex. 2: Improved MDTF Expertise and Technology

Increase of MDTF Expertise and Technology Improvements							
PE		ED		MDTF		Global Parameters	
PE Expertise	1.5	ED Expertise	3	MDTF Expertise	2	% Obs Direct to MDTF	0.2
PE Personnel	20	ED Personnel	12	MDTF Personnel	5	PED Personnel Total	25
PE Technology	1	ED Technology	1	MDTF Technology	1.5	MDTF Allocation	0.2

This would provide the predictable results of increased PEF processing rates (Figure 128) and priority targets from the MDTF (Figure 129) with reduced delays with corresponding decreases in MDTF overflow (Figure 130) and capacity utilization (Figure 6.18). However, while providing improvement to targeting and minor reduction to the total overflow ratio and does nothing to reduce the strain on the PE of the federated PED since 80% of observables are still routed through the traditional PED.

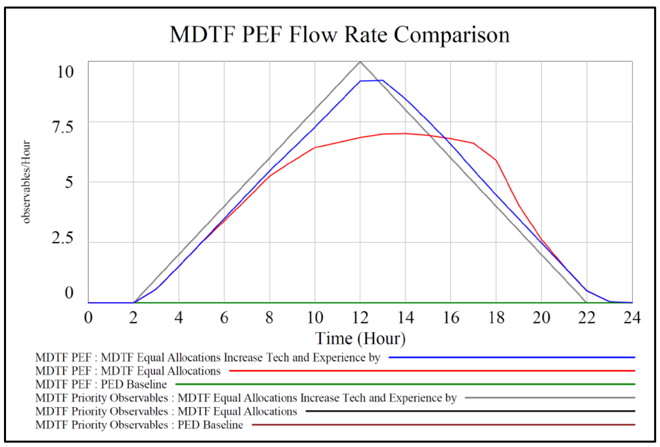


Figure 128: MDTF PEF Rate Comparison with Improved Experience and Tech

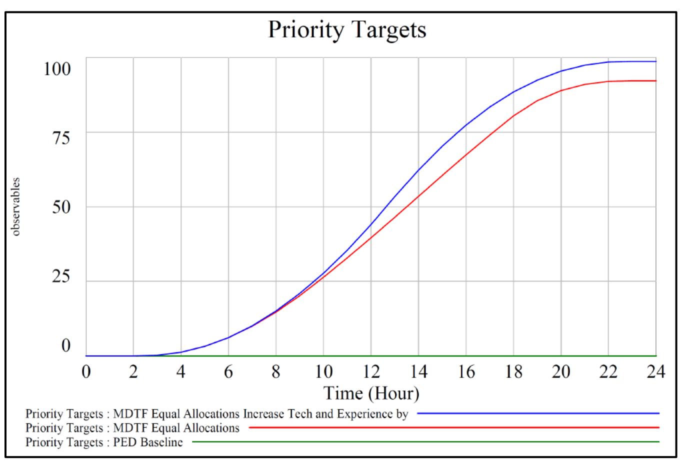


Figure 129: Priority Target Stock Comparison w/ Improved Experience and Tech

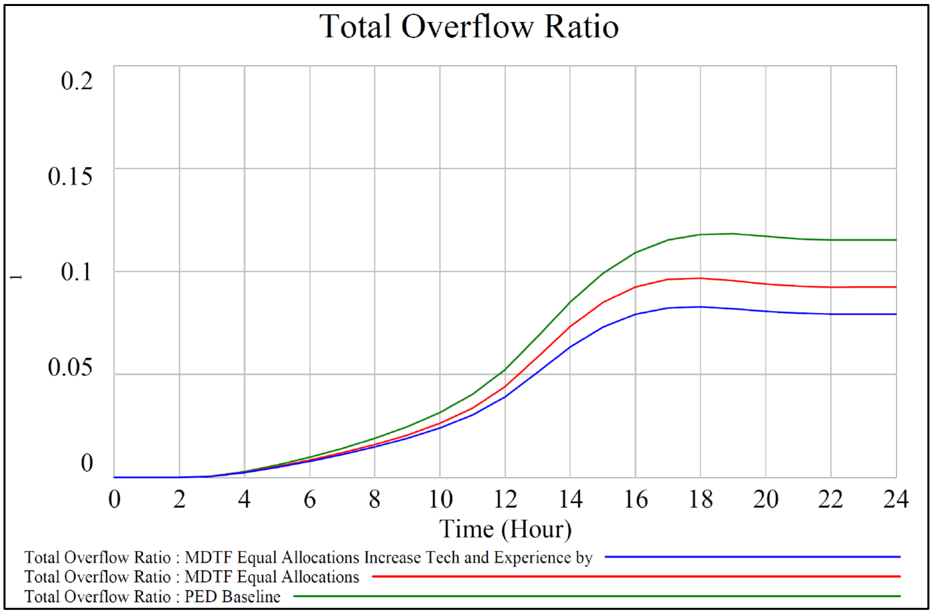


Figure 130: Total Overflow w/Increase Experience/Tech

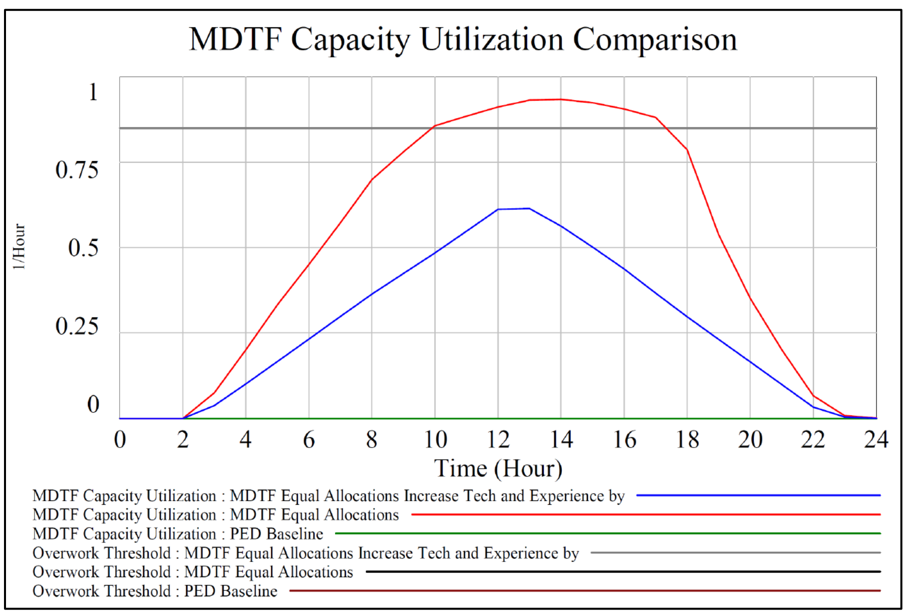


Figure 131: MDTF Capacity Utilization Comparison w/Increase Experience/Tech

How then should the manning be allocated? How much of the input observables should be routed through the MDTF? If the MDTF is increased in size to eliminate backlog and fatigue, how much technology improvement is required in the larger federated PED to prevent data loss? The possibilities are seemingly endless. In fact, for the PED subsystem model, not considering the value ranges and increments, just the combinations of which of the 10 “knobs’ or sliders to alter is 1024.

5.3.2 Sensitivity Analysis Methods

A holistic model with parametric values is commonly used in System Dynamics to ascertain how sensitive a model is to parameter values and structural changes to the model. It is used to eliminate or set constant variables that have little or no impact on the behavior of the system over time. Often this is used in SD in the iterative model creation and validation steps as one of the twelve model testing methods detailed by Sterman. Sterman states that there are three types of sensitivity analysis on an SD model: numerical sensitivity, behavioral sensitivity, and policy sensitivity.[51] There are several ways to conduct sensitivity analysis, the most common for SD being univariate and multivariate, i.e. changing one value at a time or changing multiple values at a time. This type of sensitivity analysis is useful when constructing and validating the models constituent parts which are relatively simple before being combined in a more complex fashion like Ford [143] recommends in his modeling process shown along with testing the effects of extreme values on model behavior.

Sensitivity analysis in the model construct, testing, and validation steps can be used to determine which elements are most influential on the numerical sensitivity, behavioral sensitivity, and policy sensitivity and as such can inform the model creator which variable can use estimates (little impact on model output) or need more refined data (more impact).[141] This can also be useful when defining requirement ranges and values for future systems, even if in generic terms.

Univariate and multivariate sensitivity analysis is suitable for simple models or when using the model to visually explore policy with decision makers as was previously demonstrated. However, it is slow and tedious. The more complex the model and the more variables that can be changed, leads to the curse of dimensionality, where every combination cannot be explored. Experiments must be repeatable and simply randomly changing variable for only a handful of cases limits the ability to investigate all possible combinations of potential ranges of variables.

5.3.3 Selecting Variables of Interest

Vensim has built in sensitivity analysis. For this example, it is assumed that the ED experience and manpower are required at the traditional federated PED for their final analysis and dissemination. Additionally, no advance technology factor exists to improve the ED. Hence, it is assumed that manpower must be reallocated from a fixed number of PE personnel to man the MDTF as was assumed in previous examples. Additionally, it is assumed that the traditional federated PED is required, hence, not all the manpower nor all the observables can be routed through the MDTF. The percentage of observables can be assumed to directly correlate to specific targets or via a means to prioritize inbound feeds. It is also assumed that the best possible technology can only improve productivity per analyst by a maximum factor of two. Lastly, as senior experience is required at the ED stage, it is assumed that the maximum average experience level at the MDTF and PE is two. Given these assumptions, the following ranges of values will be simulated.

Table 25: PED Subsystem Sensitivity Testing Value Ranges

PED Sensitivity Testing Value Ranges							
PE		ED		MDTF		Global Parameters	
PE Expertise	1-2	ED Expertise	3	MDTF Expertise	1-2	% Obs Direct to MDTF	0-.4
PE Personnel	15-25	ED Personnel	12	MDTF Personnel	0-10	PED Personnel Total	25
PE Technology	1-2	ED Technology	1	MDTF Technology	1-2	MDTF Allocation	0-.4

With the primary variables of interest identified and ranges of values established, the output variables of interest related to the MOP/MOE must be determined. The ultimate objective of the PED, both traditional and MDTF, is to process as many observables as possible in each period. This goal implies the minimization of the Total Overflow Ratio which is the sum of the overflow ratios for the MDTF, PED and ED. This goal of maximized output also implies the maximization of processing rates and the reduction of associated backlog delays (shifting of the processing rate curves left to attempt to match the peak of the inputs as demonstrated in Figure 120 and Figure 128). Finally, for PED decisionmakers, there is a secondary priority of reducing capacity utilization and overwork for the MDTF, PE, and ED. These will be the primary output variables of interest.

5.3.4 Monte Carlo Testing and DOE Types

Subsections 5.2.2.2 and 5.3.1 demonstrated manual methods by which assumptions about constant values can be changed and examined to assess impacts on overall model behavior. As discussed, the process is valuable for exploration by and education for decision makers but is tedious and involves altering too many variables, especially for larger problems, the curse of dimensionality. Monte Carlo simulation was previously identified as missing from the executable architecture in Subsection 2.3.2.3. This method of simulation is also known as multivariate sensitivity simulation (MVSS). It allows for the variation of all designated variables to evaluate their effects on the outputs via hundreds or thousands of simulation experiments using the “Law of Large Numbers and other methods of statistical inference [150]. With Vensim, this function is built into the executable architecture. Because the SD executable architecture is built off a series of differential equations, Monte Carlo simulations can be executed in a matter of seconds or minutes depending on the number of input and output variables. Either way, this speed provides a distinct advantage over other types of EA.

When developing the Monte Carlo simulation, several design of experiments (DOE) exist to adequately sample the multidimensional design space while decreasing processing speeds. In a traditional Monte Carlo simulation, input variable values are selected randomly within a given range based on probability distributions applied to the variables. The outputs values are obtained for every run then analyzed statistically, typically for their probability distributions, means and standard deviations [151]. The most basic DOE types are factorial sampling designs, either full or partial, but also include Packet-Burman Designs, Taguchis Orthogonal arrays, and Response Surface Methods.

As built in functionality, Vensim™ offer six methods of sampling: univariate, multivariate, Latin Hypercubes, Latin Grids, and external files. Univariate changes one input diameter independently based on the probability distribution applied while holding the other values constant. This cycle is repeated for all input variables selected. Multivariate changes all of the parameters randomly for each run based upon the probability distributions for each variable (traditional Monte Carlo). Latin Grid is a brute force systematic search of every possible combination. Even for SD, large models with many input variables and outputs of interest can be computationally expensive and slow using the Latin Grid method. The 'File' method allows users to build a DOE in external software and import it via a tab delimited text file to run in Vensim. Lastly, the Latin Hypercube is like the multivariate method, but stratifies the parameter values in each dimension. If uniform distribution is applied the input variables (which is typical without a priori knowledge of likely values in the range) "a random value is drawn in each of the N-hypercube intervals in the min-max range [142] ."

Latin Hypercubes method of DOE was used for this experiment. While the number of parameter for this PED/MDTF example were few and hence computationally inexpensive, the combined problem in the subsequent chapter is not. Latin Hypercubes offer the best "all-purpose class of designs" especially when the variables of interest are continuous and modelers have a "considerable *a priori* uncertainty about the response [152]."

5.3.5 Interpreting the Sensitivity Outputs

Latin Hypercube DOE sampling for 10,000 Monte Carlo multivariate simulations was used to evaluate the effects of inputs variables of interest on the MOP/MOE of interest defined in Subsection 5.3.3. In the interest of brevity, the discussion will only focus on three essential outputs: 'Total Observables,' 'Total Overflow Ratio,' and MDTF Overwork. The other outputs demonstrate similar results but are most clear in these outputs. Arguably these three outputs constitute the most important for the system when determining the benefit and mix of the MDTF.

5.3.5.1 Analysis of Results

The graph in Figure 132 shows the sensitivity analysis with confidence bounds for the 'Total Observables' during the 24-hour period. Overlaid on the graph are the line plots from previous manual univariate analysis from previous that can provide even further insight into the effects.

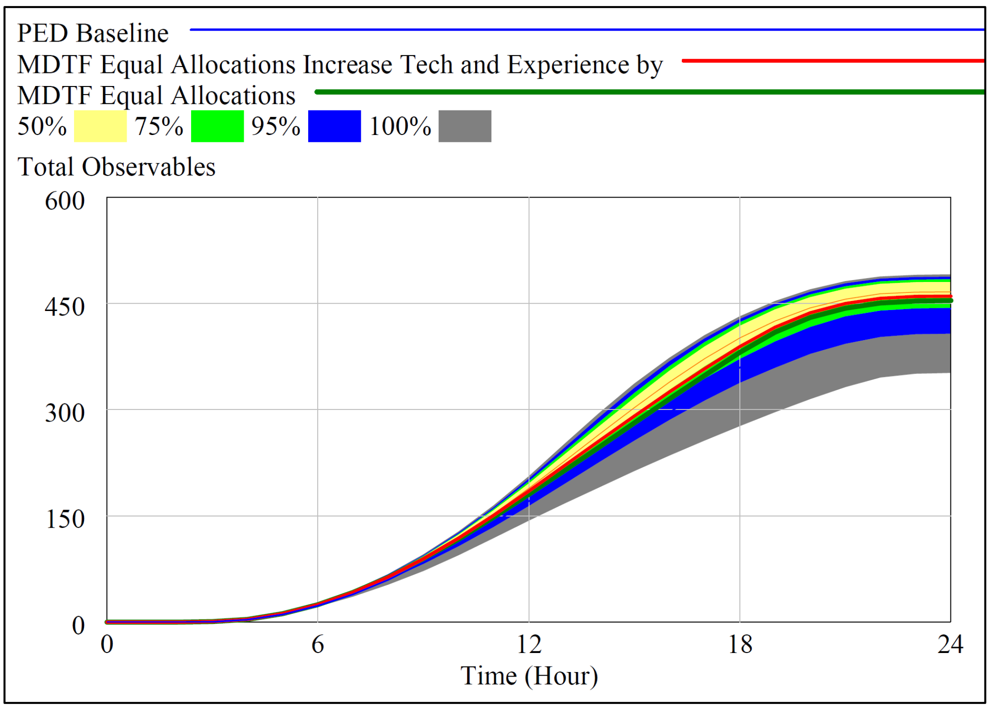


Figure 132: Total Processed Observable Sensitivity Test Confidence Bounds

As expected, regardless of the values within the selected ranges for the input variables from Table 25 there is very little variability for the first 8 hours as inputs do not exceed processing capacity of the MDTF or the PE. However, as observables begin to exceed capability depending on the selection of the inputs the variability begins to grow. While every combination leads to the same S-shaped growth pattern for consistent output behavior, by the end of the 24-hour period, the 95% confidence interval spans a cumulative output range of approximately 75 observables and the 100% spans 150 range.

Unfortunately, the confidence interval plot does not provide insight into which of the selected variables in the most influential nor the right combination of effects in the larger scheme of operations. Viewing the same results as a trace graph (showing every run as an individual line) allows the color plots of the previous manual tests to be more easily seen (Figure 133). Insights gathered from the manual manipulations can aid in the interpretation of the sensitivity analysis.

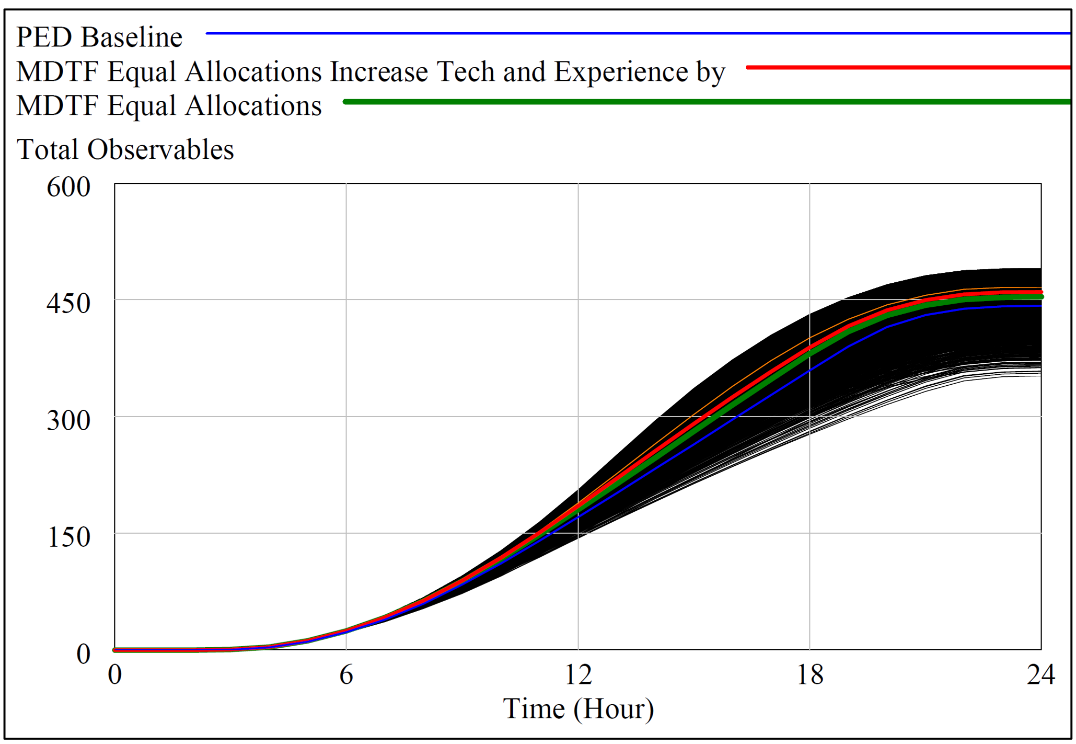


Figure 133:Total Processed Observable Sensitivity Test Individual Traces (10000)

Looking closely at Figure 133, the blue line indicates the base run (Table 20); the green line indicates the MDTF with equal allocations of personnel and observables (Table 21); the red line indicts the same MDTF allocations with improved technology and experience (Table 24). Finally, the orange indicates the average for all the 10,000 runs.

It can be observed that adding the MDTF increased the productivity, which was further increased by the increase of MDTF experience and technology. However, when comparing the magnitude of the increase, the simple addition of the MDTF (green) caused a greater increase over the baseline (blue) than the technology and experience increase (red) did over the MDTF alone (green).

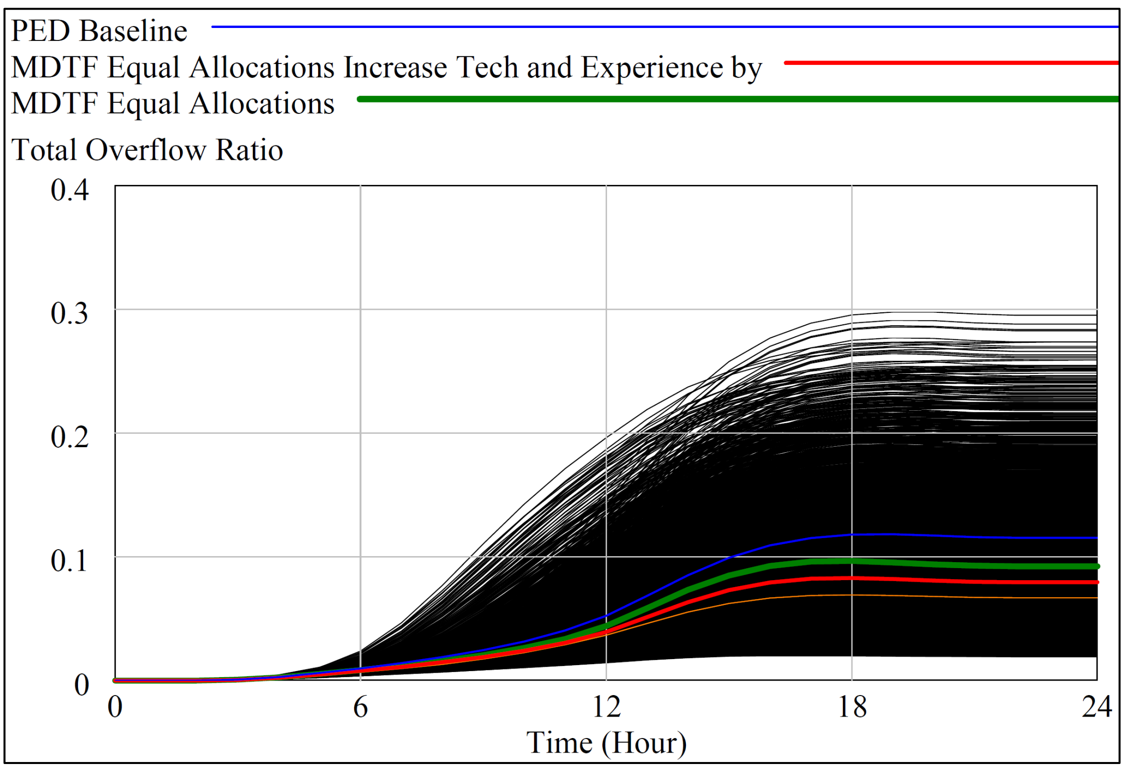


Figure 134: Total Overflow Ratio Sensitivity Test Individual Traces (10000)

Recall that Total Overflow is inversely proportional to the Total Observables; reduced values are desirable and indicated fewer lost observables. The sensitivity trace plot of the Total Overflow Ratio (Figure 134) demonstrates similar behavior in the inverse. From these observations, one can deduce that while both improvements in technology and experience improve output, the allocations of intelligence and manpower to the MDTF has a greater effect on the desired end state.

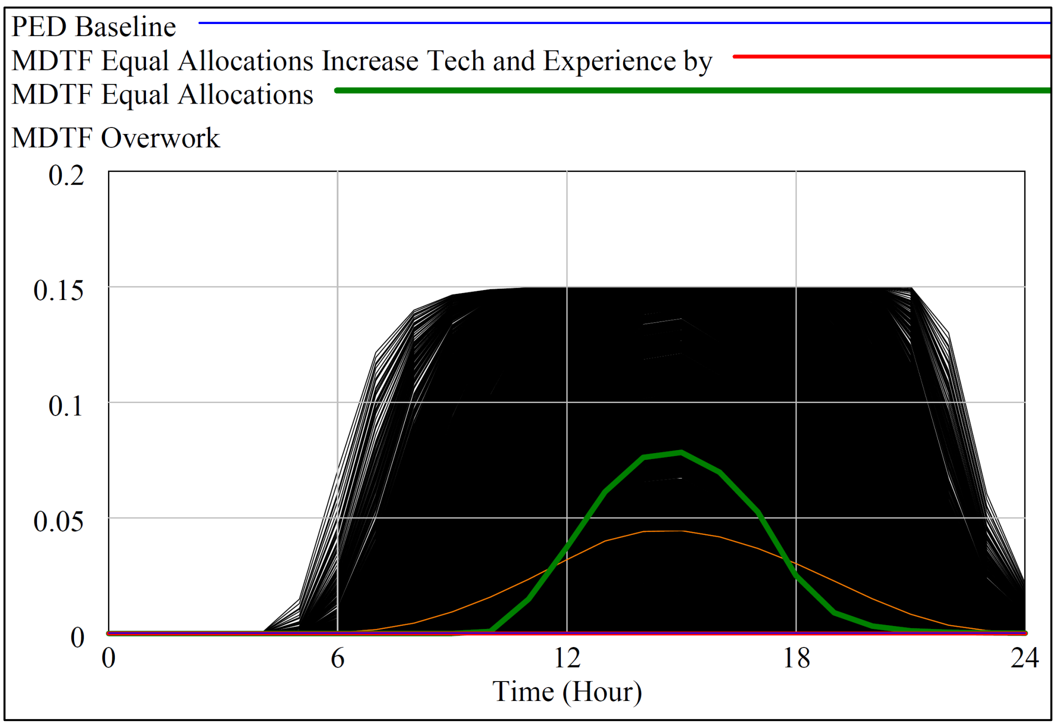


Figure 135: MDTF Overwork Sensitivity Test Individual Traces (10000)

Finally, the MDTF Overwork shows a huge variability in values over the runs (Figure 135). From first look, it provides little insight into which input variables are the most influential. However, like previously, the manual variations provide additional insight as to the effects of the input variables. For the baseline (blue) there is no overwork;

naturally, since the baseline does not include the MDTF. The equal allocation line (green) shows an overwork that is slightly greater in magnitude but has lower variance than the average (orange). The increase in technology and experience (red) however is flat, indicating that the improvements in technology and experience have the largest effect on reducing overwork.

Given that the primary mission of the PED is the production of observables, the two most influential parameters are the allocations of observables and manpower. To test this deduction, the Monte Carlo simulation can quickly be run again but by setting the allocation back to the equal allocation values (20%) and only varying the technology and experience of the MDTF. The effects are depicted in the subsequent graphs for the outputs of interest.

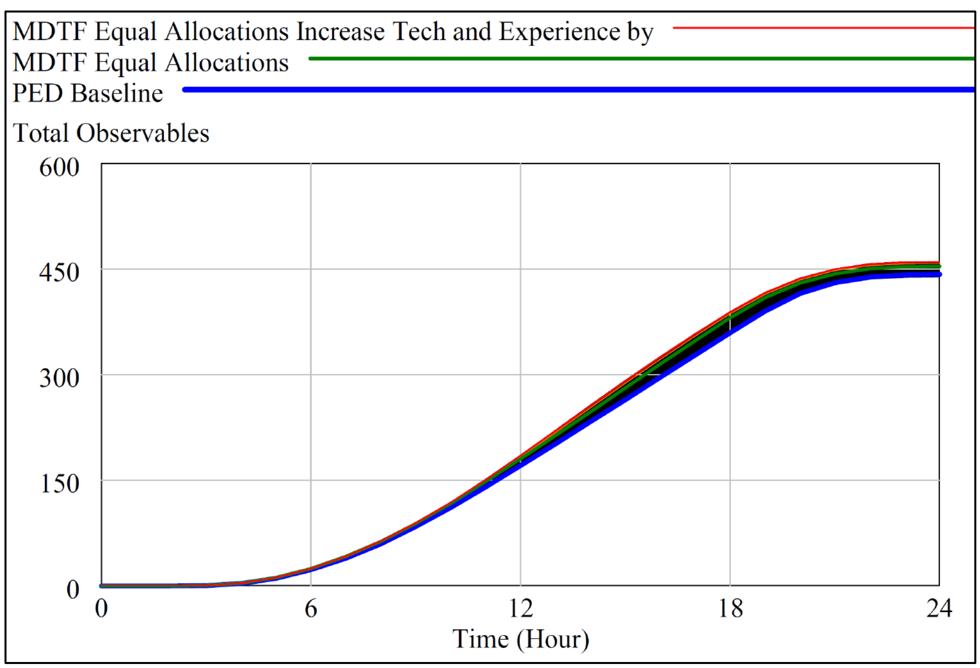


Figure 136: Total Observable Sensitivity Test - Only Vary Tech and Experience

The graphs in Figure 136 and Figure 137 compared to Figure 133 and Figure 134display a significant decrease in variability and, hence, sensitivity to the selected input parameters, as suspected. This observation confirms the assertion that PED subsystem productivity (Total Observations and Total Overflow Ratio) is more sensitive to the allocations of observables and personnel than it is to experience and technology factors. Conversely, while the variability in Figure 138 has reduced in variance compared to Figure 135, the magnitude peak remained the same. Hence, we can conclude that the experience and technology are important in reducing Capacity Utilization and Overwork.

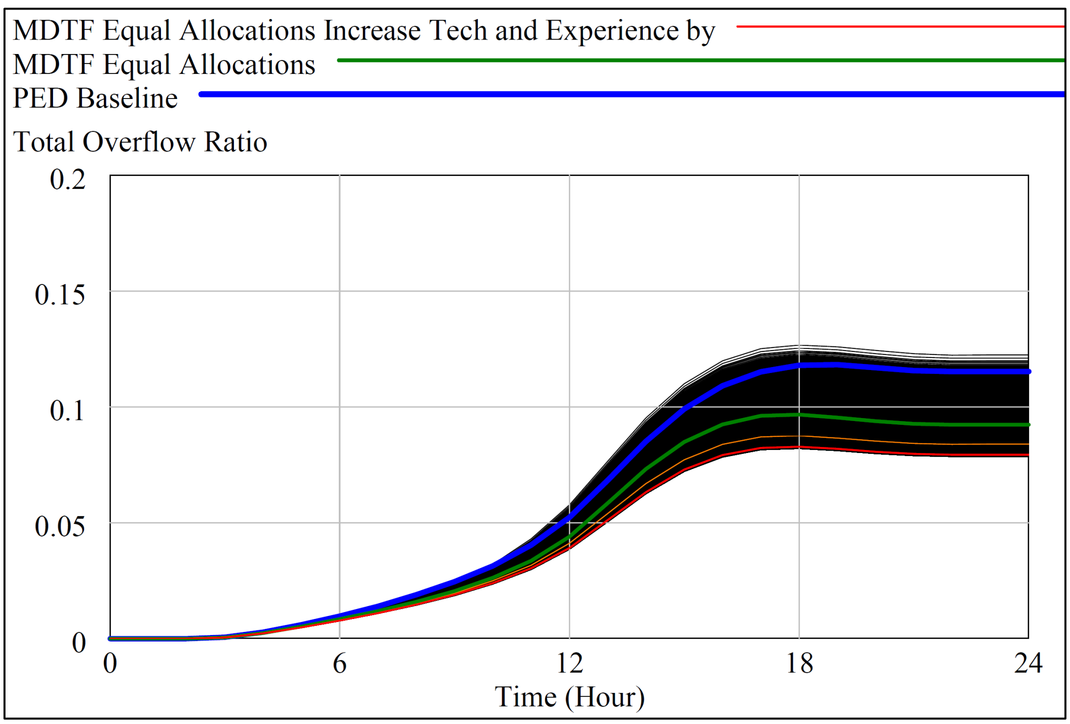


Figure 137: Total Overflow Ratio Sensitivity Test -Only Vary Tech and Experience

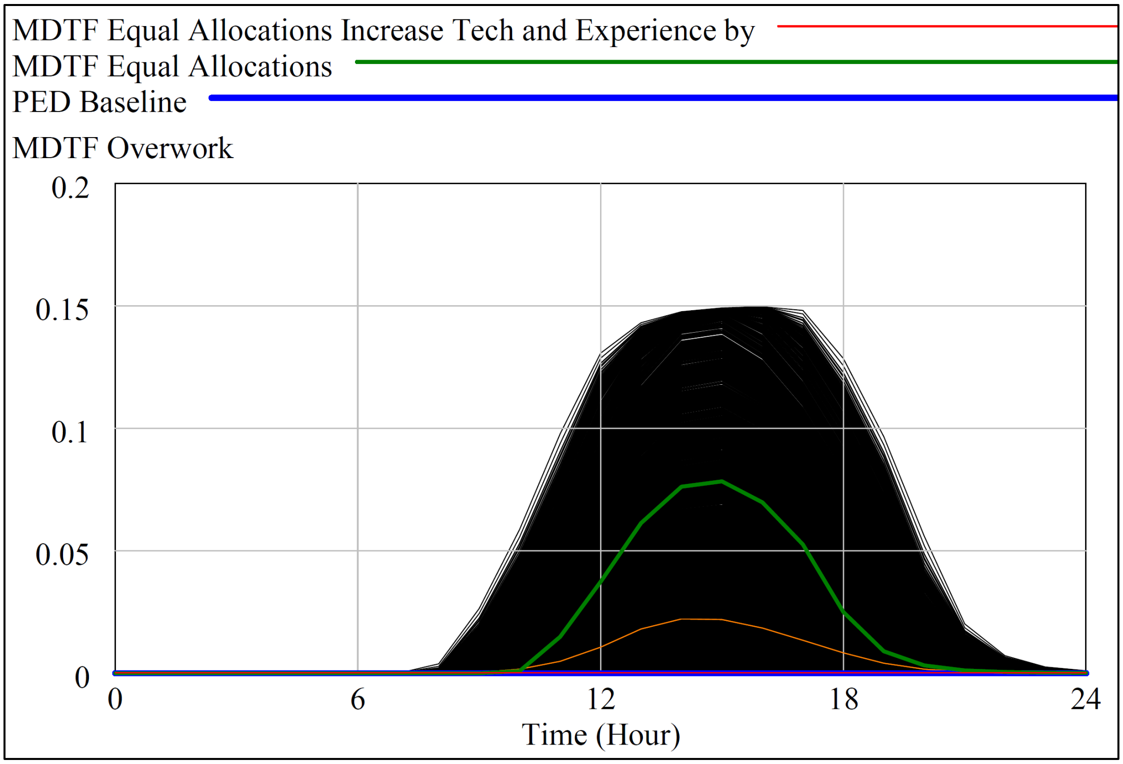


Figure 138: MDTF Overwork Sensitivity Test- Only Vary Tech and Experience

5.3.5.2 <u>Seeking an Optimal Solution</u>

The sensitivity analysis in the previous section provides important insights to decision makers when determining policy and structure as well as allocation decisions to support PED operations. Armed with knowledge gained from the sensitivity analysis, manual experiments and games may be played in the executable architecture with the ability to instantly view effects. However, the sensitivity analysis provides insight information, but little in the way of combinatorial specificity.

Fortunately, Vensim has built in a variety of robust optimization algorithms to identify the best combination of selected input variable values to maximize (or minimize) desired outputs. The standard optimization process utilizes a Powell optimizer (a derivative-free conjugate direction method), to identify the local of the minimum within a specified tolerance value or number of iterations. More advanced optimization methods are also available such as Markov Chain Monte Carlo (random walk) for calibration of input variables to match externally provided output data and Simulated Annealing for exploring payoff surfaces with multiple optima [142].

For the standard Powell optimizer, the user defines and provides weights (for scaling) for "payoff values" and selects the input values of interest. If the single objective were to maximize the output of the PED subsystem only, the answer would be trivial—simply maximize all of the technology, manning, and experience levels, while allocating 100% of the objectives through the MDTF to completely bypass the legacy federated PED architecture. Such a solution, however, does not realize the constraints of real-world operations (limited quantity of manning, skill sets, bandwidth, and workstations; increased manning in forward areas of operations at risk; lack of available technology; etc.).

From the sensitivity analysis we know that the allocations of manning and observables has the largest influence on the sensitivity of the response. We also know that training/experience and technology improves efficiency but mostly affects capacity utilization. As an example, assume that the creation of the MDTF is immediate, utilizing existing skill sets and technology. How then, should observables and manning be allocated to the provisional MDTF to minimize overflow (lost intelligence) and minimize overwork of not just the MDTF, but also of the PE and ED?

With these thoughts in mind, the following parameters were loaded into the Vensim optimization interface.

Optimization Input Parameters				
1.0	<=	PE Expertise	<=	1.5
1.0	<=	PE Technology	<=	1.0
1.0	<=	MDTF Expertise	<=	1.5
1.0	<=	MDTF Technology	<=	1.0
0.1	<=	% Obs Direct to MDTF	<=	0.4
0.1	<=	MDTF Allocation	<=	0.3
Optimization Payoff Elements			**Weight**	
Total Overflow Ratio			-1	
MDTF Overwork			-1	
PE Overwork			-1	
ED Overwork			-1	

Maximum payoff found at:
"% Obs Direct to MDTF" = 0.176431
MDTF Allocation = 0.2
MDTF PEF Technology = 1
PE Technology = 1
PE Expertise = 1.5
*MDTF PEF Expertise = 1.5
Simulations = 133
Pass = 3
Payoff = -4.17464

Figure 139: PED Optimization Input Parameters and Results

From the results, we can observe that a slightly greater amount of personnel should be allocated to the MDTF than observables to find the best balance of total overflow and overwork on all the stages. Because we are trying to optimize the best combination of four equally weighted output metrics, the output may not be the best in each category but is the best combination overall (see Figure 140, Figure 141, Figure 142, and Figure 143).

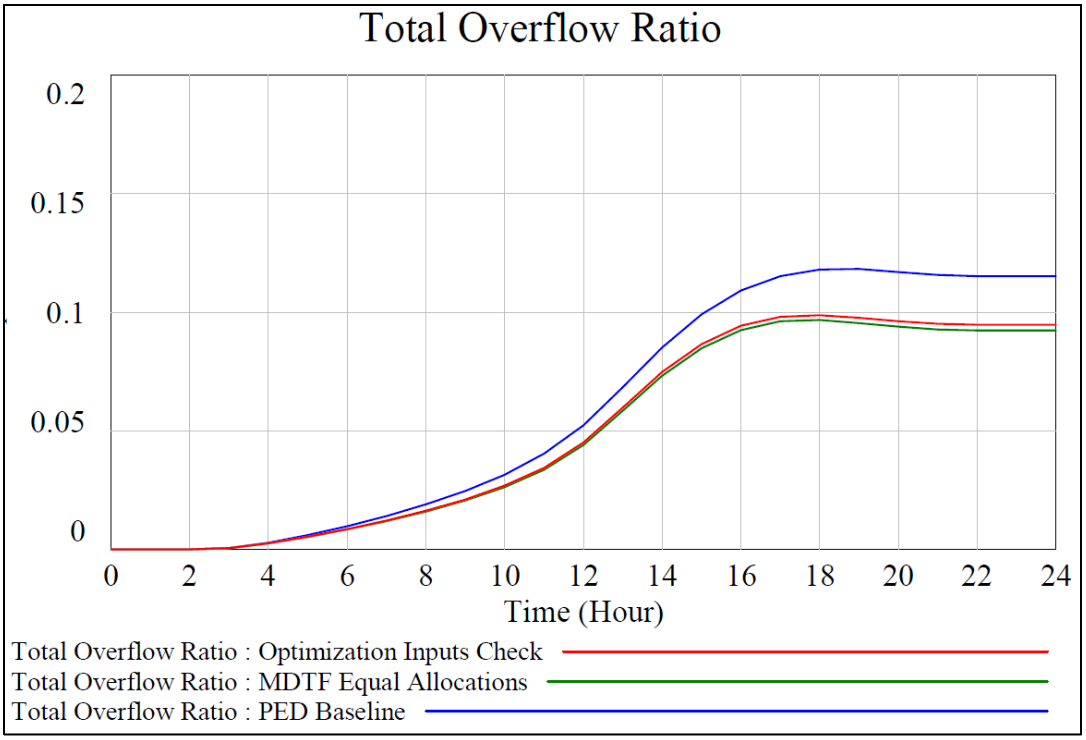


Figure 140: Total Overflow Ratio Comparison with Optimized Inputs

Figure 140 shows the marked improvement on the reduction of the overflow ratio using the optimization results values (red) compared to baseline (blue), though it is slightly higher than the pure equal 20% distribution of manning and observables (green). Figure 141 shows the reduction of peak MDTF overwork of 3.4%.

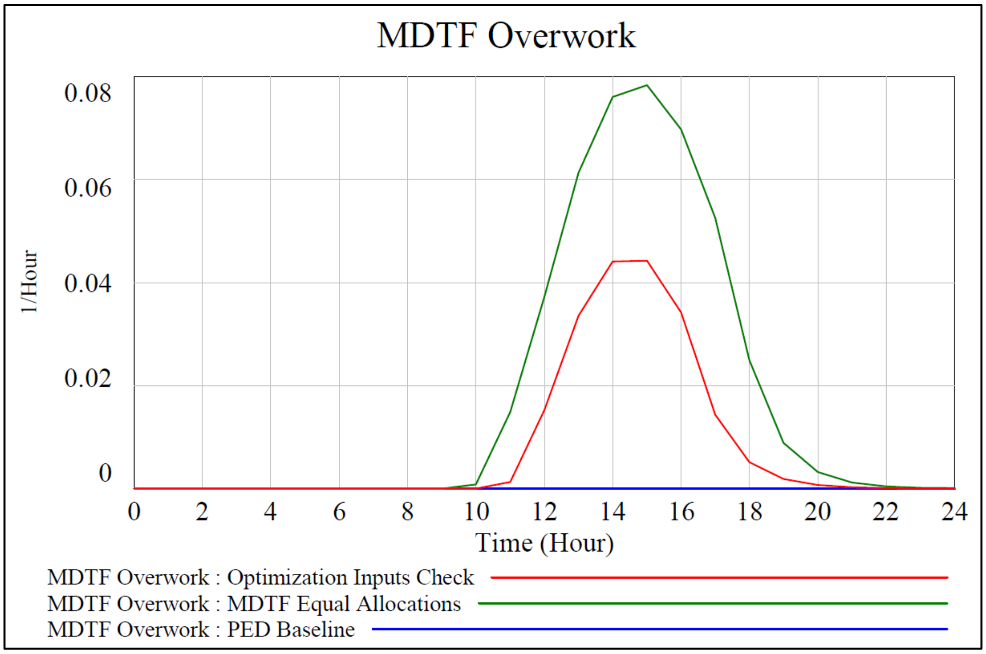


Figure 141: MD Total Overflow Ratio Comparison with Optimized Inputs

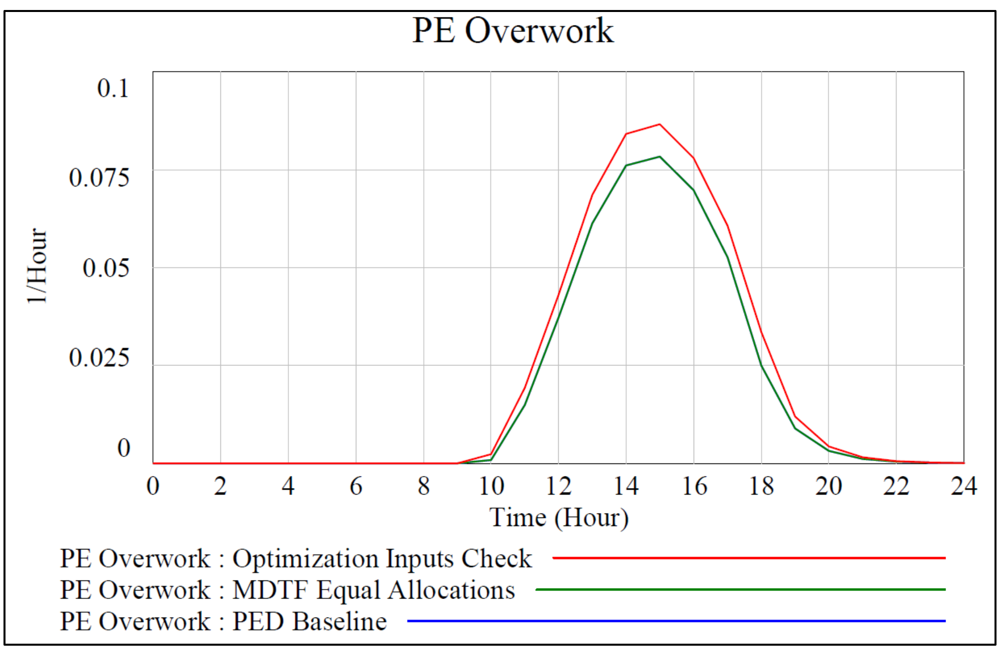


Figure 142: PE Overwork Comparison with Optimized Inputs

Figure 142 shows the small increase of 0.9% in the PE Overwork of using the optimization results values (red) compared to baseline (blue) and equal 20% distribution of manning and observables (green). Figure 143 shows the elimination of ED Overwork.

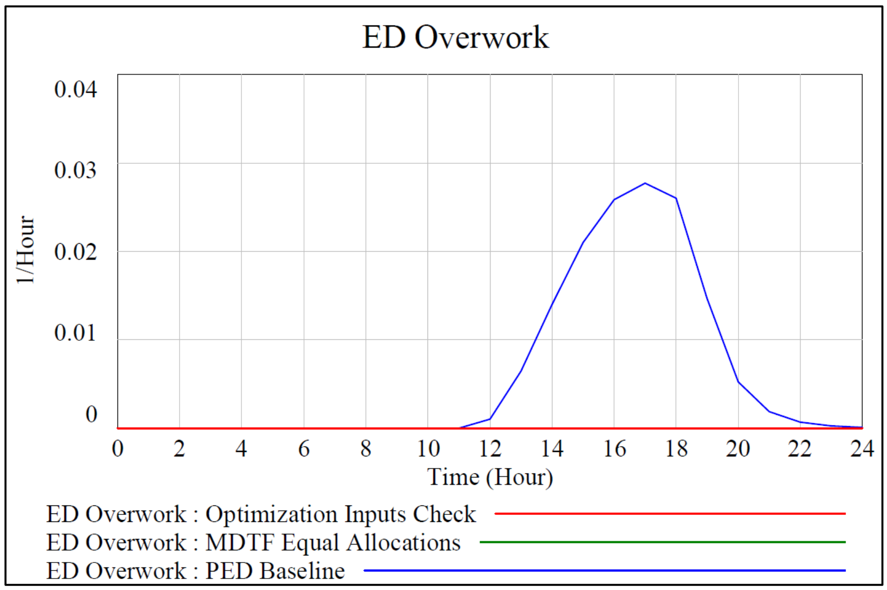


Figure 143: Total Overflow Ratio Comparison with Optimized Inputs

5.4 Summary of Findings for Research Question 2

Research Question 2 hoped to explore the ability of the executable architecture to explore the effects of structural and policy changes on the systems and the systems-of-systems. To show that the SD executable architecture can effectively evaluate such questions, the PED subsystem was utilized to explore specific proposed DOTmLPF-P changes including the addition of the MDTF, technology improvements, and training/experience improvements.

By using a standardized input, the ability to explore input variable and structural changes manually to play games and visualize effects was demonstrated using assumed value ranges. This allowed the interpretation of effects and the user to make more educated assessments as to the impact of variations. The input variables were then assessed over the entire span of ranges using Monte Carlo simulation, sped up by the utilization of Latin Hypercube DOE sampling for 10,000 combination of input variables with near instantaneous results. A method to explore the results via the aid of the initial manual methods was introduced. Finally, once the most influential variables were identified, the ability to optimize the combination given desired outputs was demonstrated using built in Vensim SD functionality.

For the specific MDTF problem, RQ2.2 asked, given new structure, how intelligence and resources should be allocated through the MDTF PED along with additional DOTmLPF-P improvement to improve overall mission effectiveness. Following the method explained above, the executable architecture was able to explore this question given immediate technology and experience values and determine that unequal distribution of observables and manpower is best to increase the output of the PED structure while reducing total overwork of personnel in the three stages. Additional training/experience and technology will moderately increase the total observables process in a given time but will have a greater impact on the reduction of overwork with a constrained number of personnel.

While the EA was successful in this realm, it is necessary to explore the effects given an operational scenario and the complete SoS to compare how these DOTmLPF-P structural and policy changes may be different in the larger context.

CHAPTER 6. PROBLEM 3: ALLOCATIONS

6.1 Problem 3 Summary

The reader is reminded of the research question this section hopes to address:

> **<u>Research Question 1.2</u>**
>
> *Can the executable architecture be used to identify elements and values of the SoS architecture that have the greatest impact on operational MOP/MOE?*
>
> **Hypothesis 1.2:** The EA can be used to generate surrogate models to identify the most important aspects and optimal values of the SoS variables against operational MOP/MOE to inform exploration and investment.

In Chapter 4, the complete executable architecture was developed for the AISR-PED SoS in a D3A scenario against the air defense and counter battery elements of a generic near-peer threat. Chapter 5 took a step into the executable architecture to explore how to evaluate structural changes to a system within the SoS. In this chapter, we return to the large context to demonstrate how to the executable architecture can be used for the larger SoS to rapidly evaluate many alternatives across the entirety of the included systems in the larger operational context.

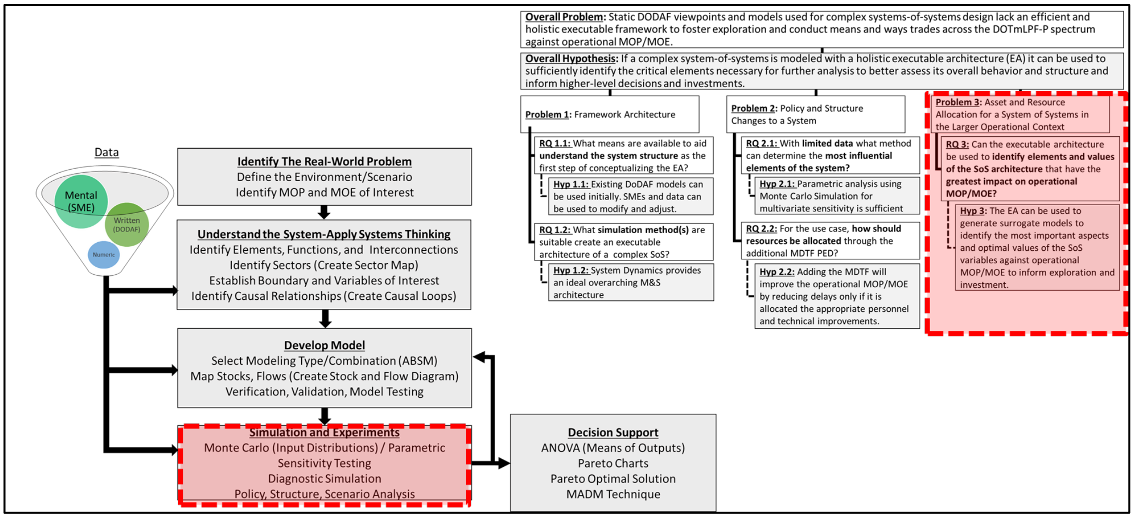


Figure 144: Problem 3 Summary

This holistic analysis technique will provide a method to make trades amongst multiple stakeholders with competing demands by determining which elements of the system are the most influential to meet the operational MOP/MOE. This method will demonstrate how to determine optimal values to best meet objects and serve as a starting point to play games, make trades, and explore alternatives and rapidly visualize the results if the assumptions and constraints used to make the model change. The Overall Problem, Hypothesis and summary of Problem 3 are summarized in the Figure 144 for the reader's convenience.

6.2 Experiment Design

6.2.1 The Interactive Dashboard

For the reader's convenience, the overall executable models are reproduced below though is a reduced scale. Refer to the figures in Figure 113 and Figure 109 respectively for larger views.

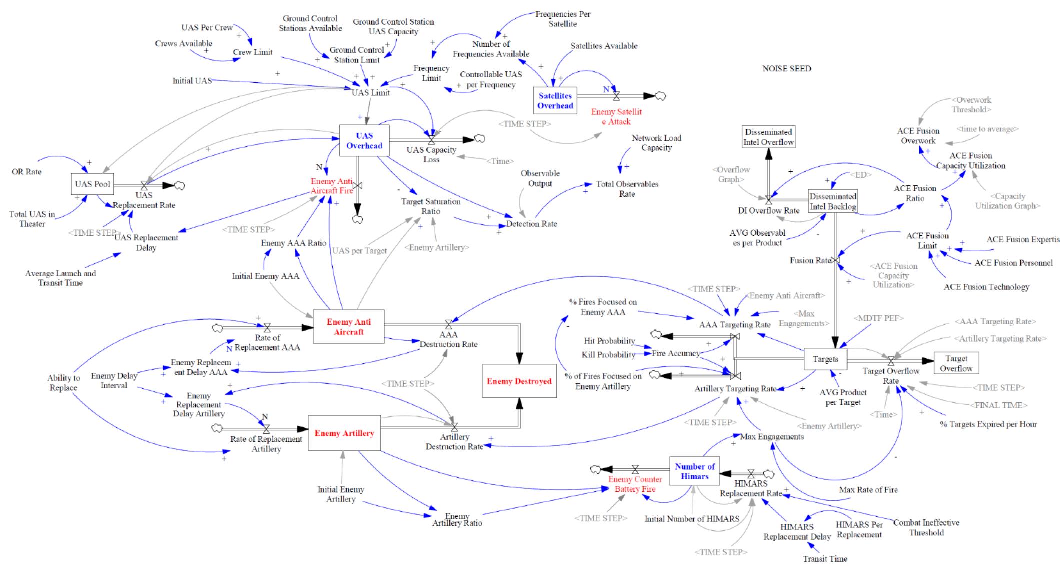


Figure 145: Overall AISR PED D3A Model

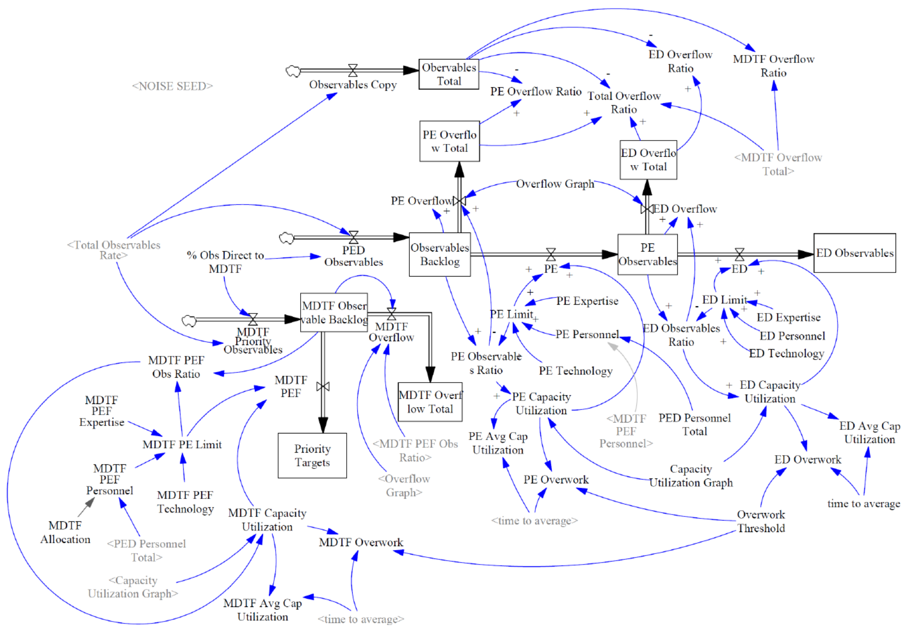


Figure 146: PED & MDTF Subsystem Model

While these executable architectures can be manipulated directly on the stock and flow diagrams via the SynthSym function in Vensim, such a view can be cluttered and overwhelming. Additionally, unless constructed as an interactive object, graphs have to be reviewed individually which can be tedious and make it difficult to visualize changes. Even if constructed and interactive graphs, the scaling on them to view such a large model makes the graphs difficult to read. Additionally, the multitude of stock and flow diagrams and variables linked by causal loop connectors can create an overwhelming, cluttered environment for stakeholder discussion and interactions. Hence a dashboard can rapidly be created with specific sliders and output graphs to play games and immediately visualize the effects. The complete dashboard is depicted in Figure 147, while Figure 148 and Figure 149 provide zoomed in views for improved clarity.

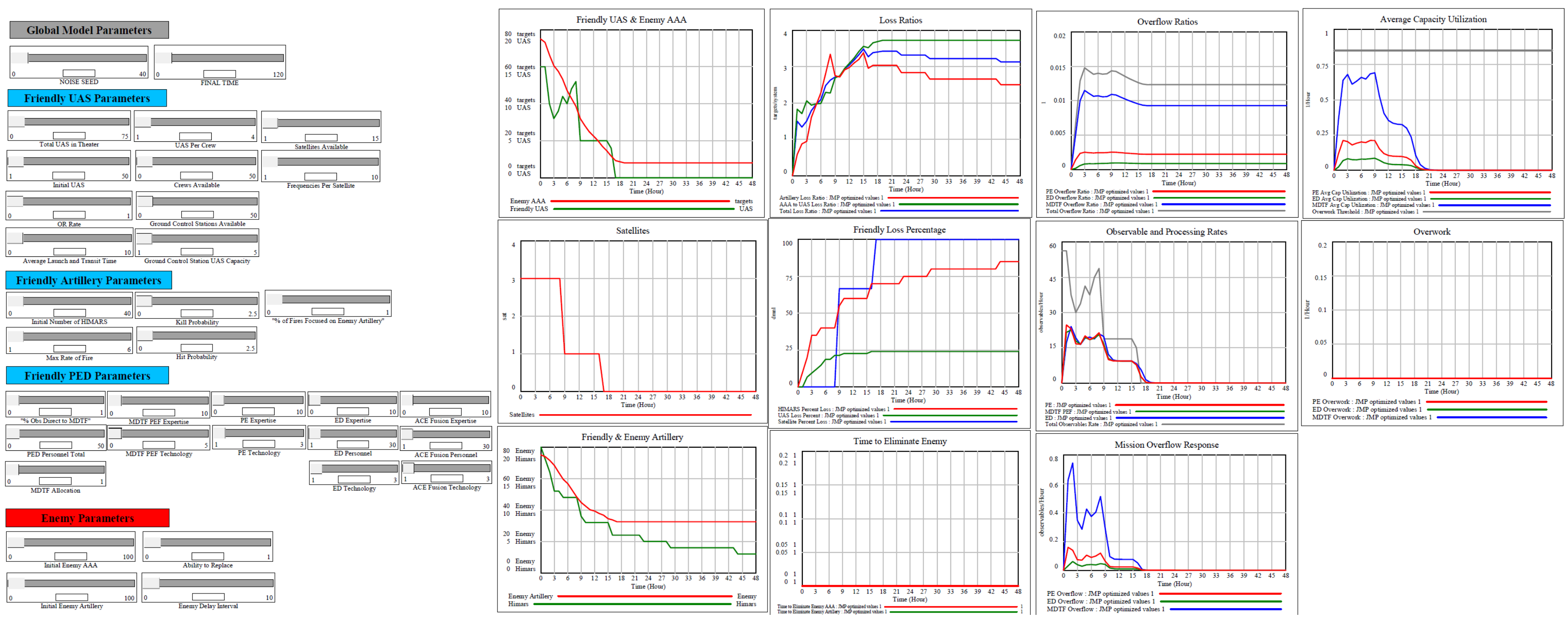


Figure 147: Complete Dashboard

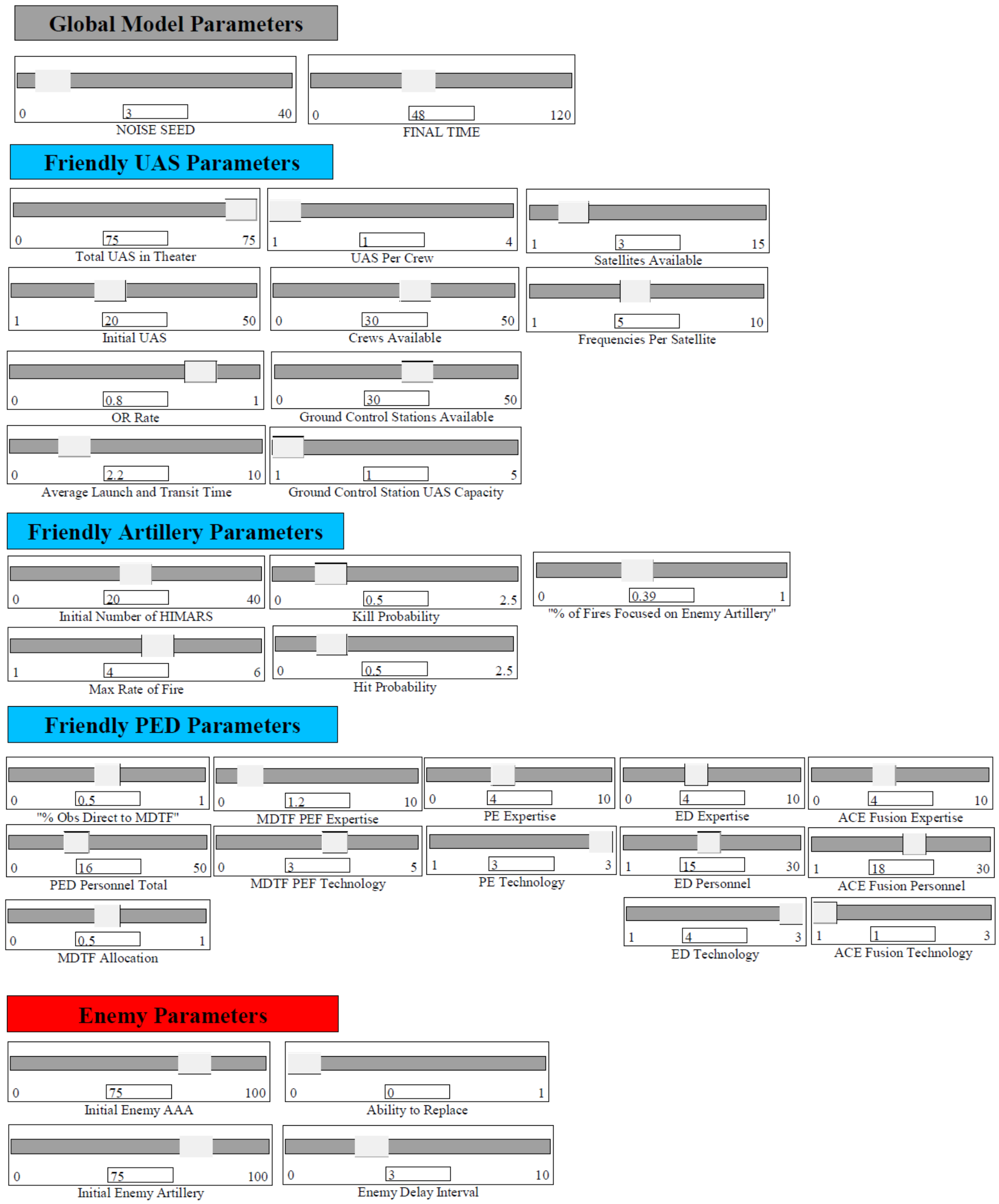


Figure 148: Dashboard Input Control Sliders

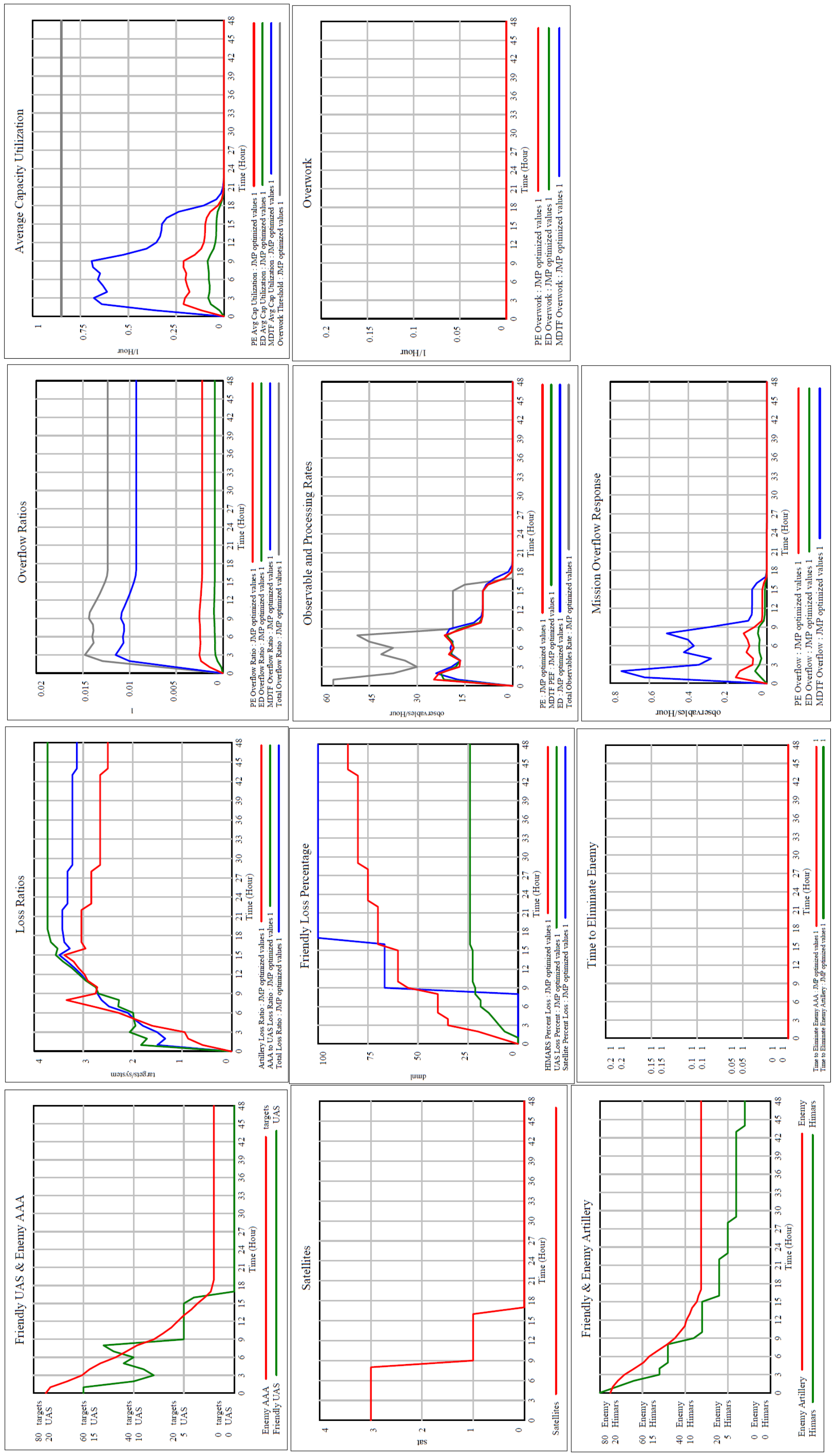


Figure 149: Dashboard Instantaneous Output MOP/MOE Graphs

6.2.2 Challenges/Observations from Games

Like with the previous problem, univariate and multivariate manual variation of variables along their ranges (particularly at the extreme values) provides valuable insight as to their effects on the operational MOP/MOE. For the reader's convenience, the operational MOP and MOE are reproduced below and are reflected in the output graphs displayed on the interactive dashboard.

Table 26: AISR-PED Measures of Performance

Friendly UAS Losses Over Time
Friendly HIMARS Losses Over Time
Friendly Satellite Losses Over Time
Enemy Anti-Aircraft Artillery (AAA) Losses Over Time
Enemy Artillery Losses Over Time
Intelligence Processing Rates
Intelligence Overflow Over Time

Table 27: AISR PED Measures of Effectiveness

Enemy to Friendly Artillery Loss Ratio Over Time
Enemy AAA to Friendly UAS Ratio Over Time
Total Loss Ratio Over Time
Friendly Loss Percentages Over Time
Intelligence Overflow Ratios
Average Intelligence Capacity Utilization
Intelligence Personnel Overwork

With this many MOP/MOE of interest, finding the appropriate combination for input variables becomes a significant challenge. This is especially true due to the number of input variables for the overall model (Table 28). Recall that with the PED subsystem model's 10 input variables that could be manipulated just considering which "switches" should be altered leads to 1024 “switch” combinations. The overall model has 41 variables. Consider just the number of combinations of “switches” that could be manipulated yields 2^41 possibilities or approximately 2.2 trillion. Bear in mind that this mind-boggling value does not even consider the ranges of the variables and the incremental divisions between them, so the actual possible combinations are much greater.

Table 28: AISR PED D3A Input Variables and Ranges

AISR PED D3A Input Parameters									
		Global Parameters					**HIMARS**		
0	<=	Noise Seed	<=	2^31	0	<=	Initial Number of HIMARS	<=	40
1	<=	Final Time *(*Simulation Duration)*	<=	72	0	<=	Combat Ineffective Threshold	<=	1
0	<=	Time Step	<=	1	0	<=	HIMARS Per Replacement	<=	4
		PED			1.0	<=	Transit Time	<=	10.0
0.0	<=	PE Expertise	<=	10.0	0.0	<=	Hit Probability	<=	1.0
1.0	<=	PE Technology	<=	3.0	0.0	<=	Kill Probability	<=	1.0
0.0	<=	MDTF Expertise	<=	10.0	0.0	<=	% Fires Focused on Enemy Artillery	<=	1.0
1.0	<=	MDTF Technology	<=	3.0			**UAS**		
0.0	<=	% Obs Direct to MDTF	<=	0.5	0	<=	Total UAS in Theater	<=	75
0.0	<=	MDTF Allocation	<=	0.5	0	<=	OR Rate	<=	1
0.0	<=	PED Personnel Total	<=	50.0	2	<=	Average Launch and Transit Time	<=	6
0.0	<=	ED Expertise	<=	10.0	1	<=	Initial UAS	<=	50
1.0	<=	ED Technology	<=	3.0	0	<=	Total Crews	<=	50
0.0	<=	ED Personnel	<=	30.0	1	<=	UAS Per Crew	<=	4
		Fusion			0	<=	Ground Control Stations Available	<=	50
1	<=	Overwork Threshold	<=	1	1	<=	Ground Control Station UAS Capacity	<=	5
1	<=	ACE Fusion Expertise	<=	10			**Satellites**		
1	<=	ACE Fusion Technology	<=	3	1	<=	Satellites Available	<=	15
1	<=	ACE Fusion Personnel	<=	30	1	<=	Frequencies Per Satellite	<=	10
0	<=	% Targets Expired per Hour	<=	1	1	<=	Controllable UAS per Frequency	<=	5
		Enemy			0	<=	Network Load Capacity	<=	1
1	<=	Initial Enemy AAA	<=	100					
1	<=	Initial Enemy Artillery	<=	100					
0	<=	Enemy Delay Interval	<=	10					
0	<=	Ability to Replace	<=	1					

This curse of dimensionality does not mean that the dashboard is not useful, however. As was demonstrated for the smaller PED subsystem in Section 5.3.1, univariate extreme value testing provides valuable insight into the influence of the variables. For example, the Initial Number of HIMARs alone has a large impact on the attrition rates of friendly and enemy assets, the difference of which could result in all enemy being destroyed in the first 48-hour or, conversely, all of the friendlies. The same holds true for the percentage of fires focused on the enemy artillery, with the remainder focused on enemy AAA. This variable explores the effect of ways, chiefly it explores the question *How should fires be focused maximize the enemy to friendly loss ratio? Is it better to eliminate all of the enemy AAA first to preserve friendly UAS that are critical to providing intelligence for targeting?* or *Is it better to eliminate the enemy artillery first to prevent counterbattery fire from destroying our artillery necessary to penetrating the enemy defenses?*

Univariate testing also demonstrates the major impact the enemy actions have on the MOP/MOE for friendly forces. Particularly, the NOISE SEED, that was discussed in Section 4.6.1. This parameter changes the random Poisson Distribution draw (in a repeatable manner) for the enemy attacks on friendly assets.

Lastly, manual multivariate changes also yield useful insight into the effects of variables. For example, nine variables influence the UAS Limit which dictates the maximum number of UAS that can be overhead. Adjust parameters individually does nothing if other variables set the limit. For example, increasing crews to the maximum does not yield any effects on the model if the Frequency Limit due to satellites already restricts the number of UAS (see Figure 98 and Figure 94).

6.3 Determining Sensitivity/Most Influential Factors

6.3.1 Reducing Dimensionality with Systems Thinking

Part of the appeal of using SD as the selected modelling and simulation paradigm was the ability to learn about the system and yield valuable insights during the development of the executable architecture framework. The constraints in the UAS Limit example demonstrates the benefit of these insights to make informed changes on the dashboard. However, to utilize the executable architecture to make decisions and evaluate means and ways alternatives, dimensionality should be reduced. The first step is manual testing from the previous section. Second is an evaluation of which variables can be influence by stakeholders and which are outside of their control. For example, continued replacements of enemy from an endless pool will always result in the loss of our limited HIMARS fleet in the MDTF Brigade (and any limited replacements from adjacent units in the reserve pool if assumed).

Hit Probability and Kill Probability are two additional variables that can make a significant impact on the behaviour of the simulation. However, when comparing the effects in the larger SoS context the author feels it is best to use heuristic values since there is no interactive simulation between assets. Fixing these variables to set values allows for easier comparison of other variables unless specifically trying to answer a question as the benefits of more accurate hits and kills, at which point all other variables could be fixed values. Alternatively, data or a smaller interactive ABM simulation could be used to refine the overall SD executable architecture if desired and time permitting (beyond the scope of this thesis)

Using these techniques, the input variables of interest were reduced to 25 (see Table 29) which is still over 35 million "switch" combinations. However, it does aid in narrowing down the focus areas for force-on-force engagements. The selected input variables also exercise the relationships in the model and prevent trivial, unrealistic scenarios that do not benefit the comparisons such as endless enemy and friendly HIMARS replacements. These variables will better highlight how to focus fires and balance kinetic and intelligence operations and capabilities. Endless reinforcement would only serve to skew numbers. The UAS replacements via the UAS Pool remain because the maximum is fixed and represents the UAS spending and acquisition. Finally, the selected variables allow us to expand upon what was previously determined for the PED subsystem, but now in the larger context of the SoS operations with variable observable input flow rather than an assumed input pattern. These variables and ranges provide a good start point for tabletop wargames and discussion with SMEs to provide insight. *But what about for analysis? How can improved combinations it be found easier and quicker?*

Table 29: Reduced AISR PED D3A Input Variables and Ranges

		AISR PED D3A Input Parameters							
		Global Parameters					**HIMARS**		
0	<=	Noise Seed	<=	2^31	0	<=	Initial Number of HIMARS	<=	40
		PED			0.0	<=	% Fires Focused on Enemy Artillery	<=	1.0
0.0	<=	PE Expertise	<=	10.0			**UAS**		
1.0	<=	PE Technology	<=	3.0	0	<=	Total UAS in Theater	<=	75
0.0	<=	MDTF Expertise	<=	10.0	2	<=	Average Launch and Transit Time	<=	6
1.0	<=	MDTF Technology	<=	3.0	1	<=	Initial UAS	<=	50
0.0	<=	% Obs Direct to MDTF	<=	0.5	0	<=	Total Crews	<=	50
0.0	<=	MDTF Allocation	<=	0.5	0	<=	Ground Control Stations Available	<=	50
0.0	<=	PED Personnel Total	<=	50.0			**Satellites**		
0.0	<=	ED Expertise	<=	10.0	1	<=	Satellites Available	<=	15
1.0	<=	ED Technology	<=	3.0	1	<=	Frequencies Per Satellite	<=	10
0.0	<=	ED Personnel	<=	30.0					
		Fusion							
1	<=	ACE Fusion Expertise	<=	10					
1	<=	ACE Fusion Technology	<=	3					
1	<=	ACE Fusion Personnel	<=	30					
		Enemy							
1	<=	Initial Enemy AAA	<=	100					
1	<=	Initial Enemy Artillery	<=	100					

6.3.2 Accounting for Noise and Many Variables

6.3.2.1 Building the DOE for Stochasticity

As with Research Question 2, the variables, ranges, and outputs of interest can be selected in Vensim to build a Latin Hypercube DOE to run a Monte Carlo simulation. The most influential variables can be identified through detailed comparison of sensitivity graphs and the gradual elimination of variables. However, with 25 variables this becomes a daunting task, if not fruitless.

Furthermore, in Problem 2, the input to the PED subsystem model was a predicted standardized deterministic shape. The larger executable architecture includes stochastic enemy-effects rates. Therefore, the number of enemy targets, UAS, and the observable rates vary with respect to time, as do the available HIMARS to prosecute targets. While the PED subsystem model had some feedback, it was minimal. The holistic executable framework demonstrates the full potential of feedback. Recall, for repeatability, the RANDOM POISSON and RANDOM BINOMIAL functions utilize a NOISE SEED variable. Manual changes using the dashboard demonstrated that the selected noise seed string had a significant influence on the responses. How then do we account for the noise to ensure our solutions are robust against unpredictable enemy effects?

While Vensim allows for the inclusion of the NOISE SEED variable as an input that can be varied over a range in the DOE, this is not particularly useful as it only adds another dimension to the n-dimensional Latin Hypercube design space. This means that the noise seed value will also get sampled with the other variables. What we desire is the average values of the outputs for every point in the DOE so that we can attempt to fit predictive models regardless of the noise variable selected and add a degree of robustness to the solution.

As was demonstrated previously, a Latin Hypercube DOE was built in Vensim without noise to create 2500 points of input variable combinations in the design space. The DOE was then exported to a custom-built MATLAB script which replicated the DOE points for 410 different noise variable values. This new 1.025,000-point DOE was imported back into Vensim using the “file” DOE option to calculate the outputs for the outputs of interest (MOP/MOE). This Monte Carlo simulation took approximately 30 minutes to run on a laptop computer, demonstrating the benefit of using SD for rapid executable architecture analysis. The results were again exported to another MATLAB script that averaged the output values for the 2500 identical points for each of the 410 noise values. The result is the same 2500 points in the input design space whose outputs are now the average of the 410 noise variables, thus providing robustness.

6.3.2.2 Using Statistical Software to Gain Insight

Now that the Monte Carlo simulation was run, a need still existed to determine which combinations help us achieve the desired results. The 2500 points with averaged output values was exported to a statistical analysis software program (JMP) to help further narrow down the most influential inputs (sensitivity analysis) and determine optimal (or at least improved) settings.

Figure 150 shows a scatterplot matrix of all the multivariate input values. Black dots represent the design points from 2500 averaged point DOE. The black, orange, and green dots represent points that will be used as training points, validations points, and testing points, respectively, for the development of surrogate models. The designations of these points were done randomly via the JMP software. This scatterplot serves to provide insight into correlation and dependency between inputs variables. Because no clear pattern emerges for the entire range of values, no influential correlations are observed. Additional scatterplot matrices can be made comparing the inputs to outputs to give us an idea of how the ranged of input values effect the magnitude of the outputs of interest (see Figure 151). Understanding the relationships via inspection expedites the selection of input ranges for which decision makers have the ability in a real-world scenario to manipulate as part of wargaming and planning. Large gaps, triangular patterns, etc. are indications of dependent relationships. The more discernable dependencies, the more likely the machine will be able to develop surrogate models.

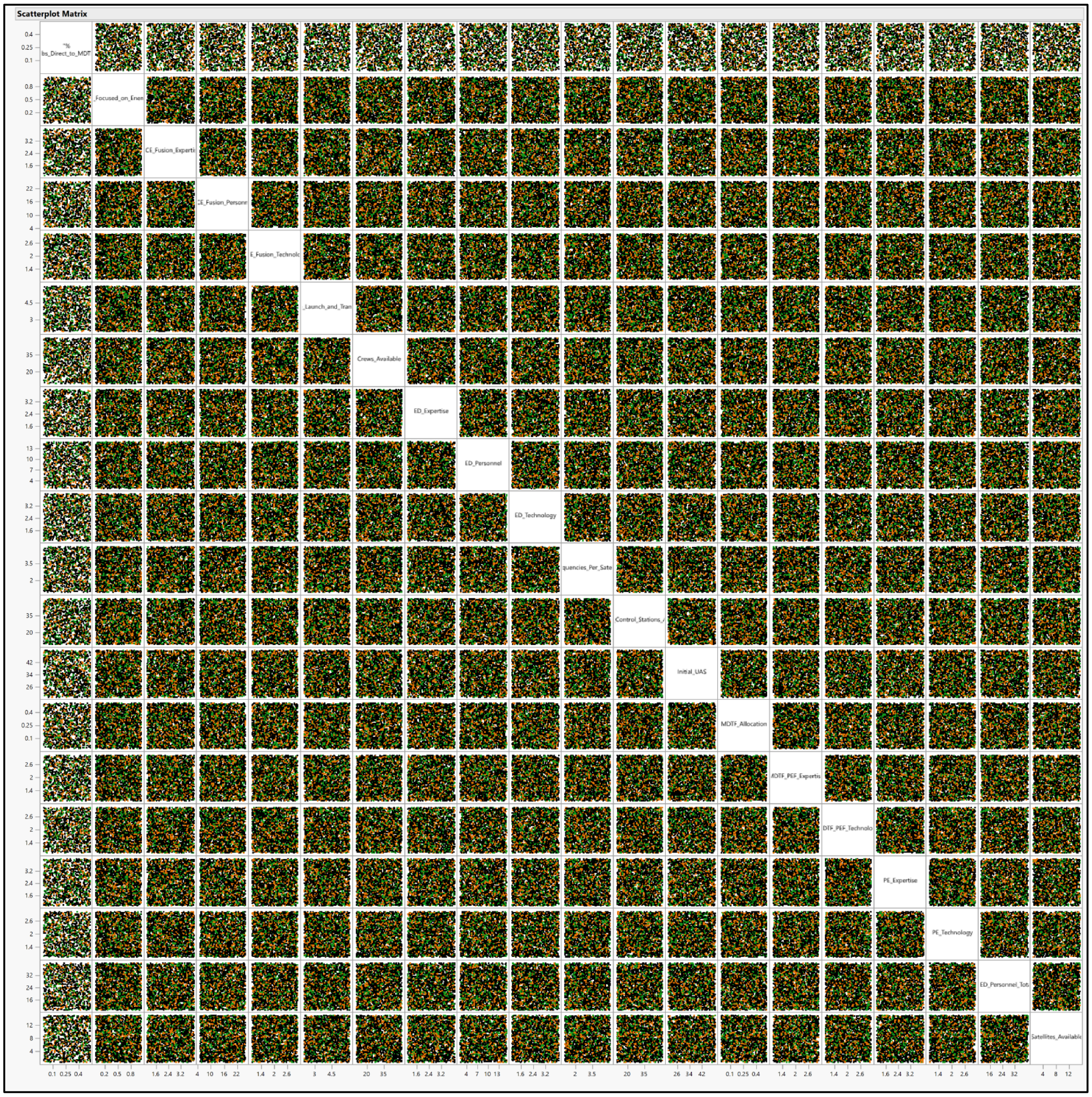


Figure 150: Scatterplot of Input Variables

Viewing Figure 151 first, a few dependencies become relevant for the input ranges and PED MOP/MOE. Not by coincidence, these dependencies reinforce the selection of key variables of interest and dependencies observed in the sensitivity analysis for Problem 2 in 4.6. The range of values for the many of the input variables has little influence on the ‘ACE Fusion Overwork’, ‘PE Overflow,’ and ‘PE Overwork Ratio.’ However close inspection indicates trends of direct dependencies of ‘Frequencies Per Satellite’, Satellites Available, with ‘ED Overflow’, which is logical since more satellites mean more aircraft overhead and, thus, more observables. ‘ED Overflow’ also demonstrates an inverse relationship with ‘ED Technology’ which is logical; more technology means more efficient processing and less lost intelligence. The most obvious correlations on the figure are inverse relation of the ‘% Obs Direct to MDTF’ to the ‘PE Overflow Ratio’ and ‘ED Overflow Ratio.’ As well its directly proportional relationship to ‘MDTF Overflow Ratio’ and ‘MDTF Overflow Total’. ‘ED Overflow’ also shows an inverse relationship with ‘% Fires Focused on Enemy Artillery’ meaning the more fires focused on artillery the less focused on AAA means more UAS shot down and less intelligence; hence, less overflow PE and ED.

Figure 151 Scatterplot Matrix of PED MOP/MOE vs. Inputs

Viewing Figure 7.9 for combat MOP/MOEs, not many clear relationships can be observed for most of the input and output values. However, four of the input variables demonstrate clear correlations. Unlike the PED, the combat MOP/MOE have yet to be unexplored in the greater SoS. Therefore, this scatterplot, provides some valuable insights.

Starting from the first column from the left, "% Obs Direct to MDTF" the following relationships are observed: The "% Obs Direct to MDTF" is directly proportional to the Total Loss Ratio, AAA Loss Ratio, and the Artillery Loss Ratio. "MDTF Allocation," Column 14 shows the same trends. This tells us that the ratio of enemy destroyed to friendly destroyed increases as the percentage of observables that flow through the MDTF increases, provided the associated manpower allocated increases as well. This confirms findings from the PED subsystem model. This is likely due to the bypassing of delays allowing a more rapid execution of kinetic effects on target.

The second column of Figure 152, "% of Fires Focused on Enemy Artillery" also presents some interesting relationships that are a bit more difficult to discern but are worth mentioning. As the "% of Fires Focused on Enemy Artillery" increases so too does the Enemy Loss Ratio (enemy artillery destroyed versus friendly HIMARs destroyed). This result is expected, but interestingly, this input variable does not show the same relationship for the 'Enemy Artillery Percent Loss' or 'AAA Percent Loss' so some another variable is coupling and influencing those outputs. The increase of focus on enemy artillery is inversely proportional to the HIMARS losses, the 'HIMARS Percent Loss' and the 'AAA to UAS Loss Ratio'. Lastly, the 'Target Overflow' shows interesting behavior with respect to the artillery focus: too little focus on artillery results in heavy overflow as does too much but remains low for most of the fires focus range.

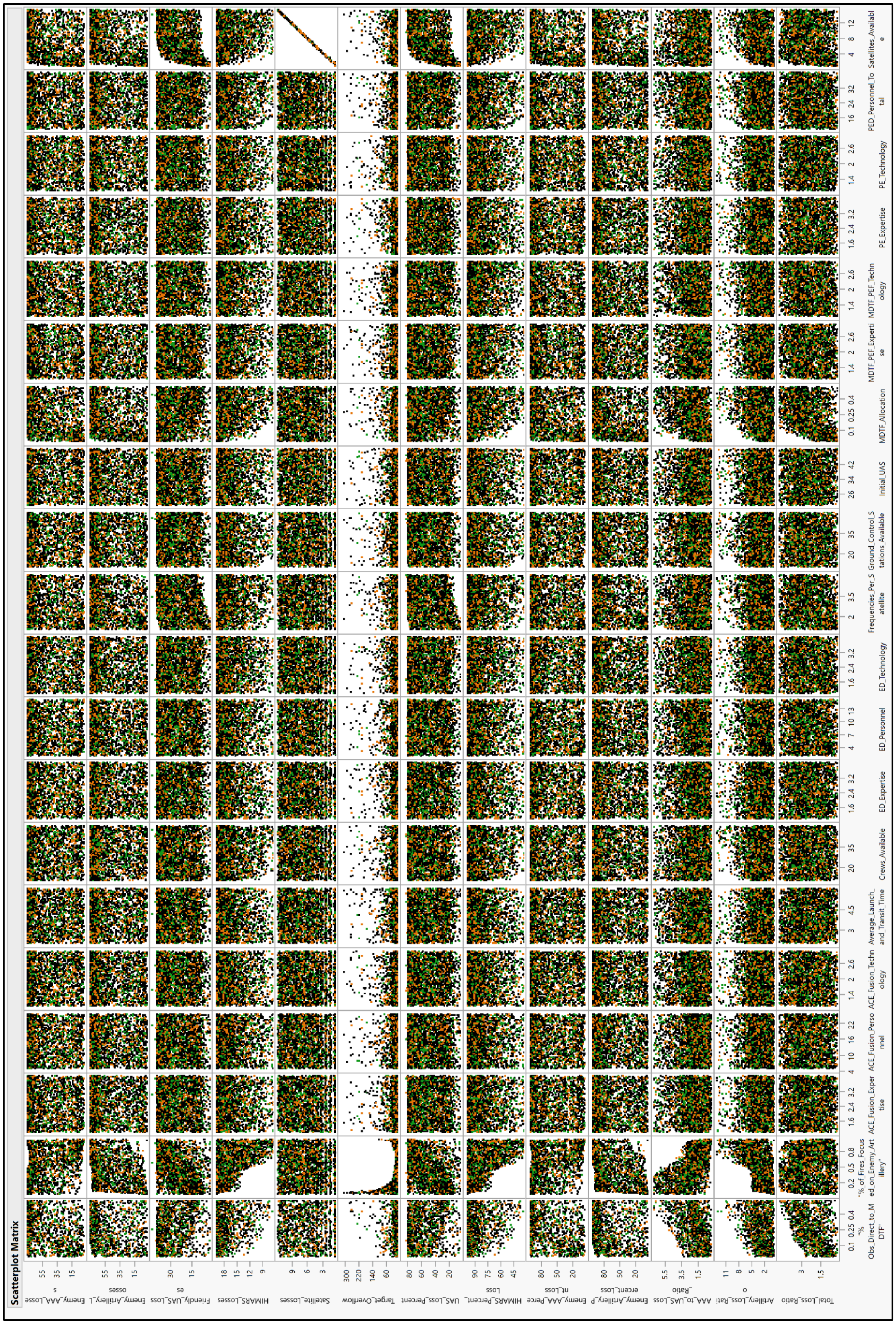


Figure 152: Scatterplot Matrix of Combat MOP/MOE to Inputs

JMP also offers a Predictive Profiler that determines the most influential variables (Figure 153 and Figure 154). The results of this profiler can be checked against the observations from the scatterplot matrices.

Predictor Screening

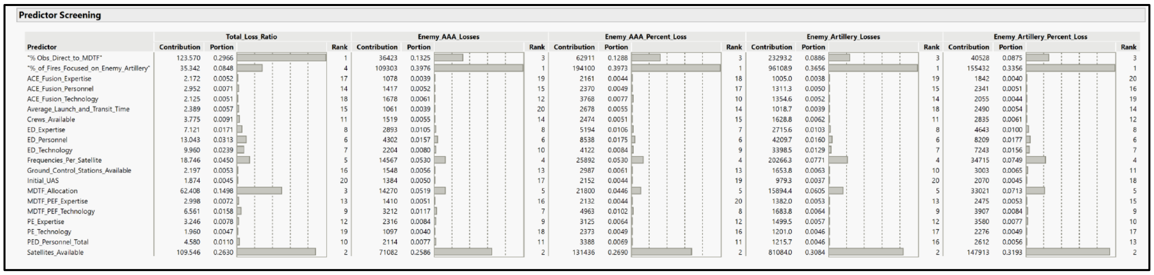
	Cumulative_Overwork			Total_Overflow_Ratio			PE_Overflow			PE_Overflow_Ratio			ED_Overflow		
Predictor	Contribution	Portion	Rank	Contribution	Portion	Rank	Contribution	Portion	Rank	Contribution	Portion	Rank	Contribution	Portion	Rank
"% Obs_Direct_to_MDTF"	40.287	0.0231	10	0.29085	0.0394	6	0.054393	0.0574	5	0.038075	0.0631	6	0.76534	0.0389	7
"%_of_Fires_Focused_on_Enemy_Artillery"	80.557	0.0462	6	0.05543	0.0075	15	0.353301	0.3729	1	0.009403	0.0156	11	6.44469	0.3273	1
ACE_Fusion_Expertise	14.034	0.0080	16	0.05365	0.0073	16	0.011869	0.0125	13	0.002227	0.0037	20	0.15811	0.0080	17
ACE_Fusion_Personnel	9.263	0.0053	20	0.03917	0.0053	20	0.041223	0.0435	8	0.005824	0.0096	14	0.21272	0.0108	13
ACE_Fusion_Technology	17.358	0.0099	12	0.05121	0.0069	18	0.012100	0.0128	12	0.003587	0.0059	17	0.75354	0.0383	8
Average_Launch_and_Transit_Time	12.335	0.0071	17	0.04734	0.0064	19	0.024682	0.0261	11	0.005982	0.0099	13	0.27542	0.0140	11
Crews_Available	11.706	0.0067	18	0.10348	0.0140	9	0.003957	0.0042	20	0.017305	0.0287	7	0.13164	0.0067	19
ED_Expertise	101.076	0.0579	5	0.71865	0.0973	5	0.067423	0.0712	3	0.002730	0.0045	19	0.83347	0.0423	6
ED_Personnel	255.997	0.1467	2	2.29623	0.3109	1	0.004778	0.0050	19	0.012262	0.0203	9	1.56282	0.0794	3
ED_Technology	134.012	0.0768	4	1.20154	0.1627	2	0.006545	0.0069	17	0.004849	0.0080	15	4.45009	0.2260	2
Frequencies_Per_Satellite	43.934	0.0252	9	0.27375	0.0371	7	0.010732	0.0113	15	0.016701	0.0277	8	1.35730	0.0689	4
Ground_Control_Stations_Available	14.054	0.0081	15	0.09280	0.0126	11	0.024871	0.0263	10	0.011384	0.0189	10	0.32684	0.0166	10
Initial_UAS	9.681	0.0055	19	0.05906	0.0080	14	0.005690	0.0060	18	0.003463	0.0057	18	1.00983	0.0513	5
MDTF_Allocation	677.660	0.3882	1	0.84492	0.1144	3	0.043839	0.0463	6	0.038999	0.0646	5	0.18652	0.0095	16
MDTF_PEF_Expertise	44.586	0.0255	8	0.07991	0.0108	12	0.028877	0.0305	9	0.003698	0.0061	16	0.23374	0.0119	12
MDTF_PEF_Technology	21.508	0.0123	11	0.05213	0.0071	17	0.007549	0.0080	16	0.007086	0.0117	12	0.20426	0.0104	15
PE_Expertise	16.825	0.0096	13	0.09469	0.0128	10	0.042996	0.0454	7	0.147916	0.2450	2	0.20777	0.0106	14
PE_Technology	14.639	0.0084	14	0.06942	0.0094	13	0.010940	0.0115	14	0.049593	0.0821	4	0.15168	0.0077	18
PED_Personnel_Total	80.008	0.0458	7	0.21362	0.0289	8	0.129802	0.1370	2	0.166504	0.2758	1	0.32986	0.0168	9
Satellites_Available	146.020	0.0837	3	0.74875	0.1014	4	0.061856	0.0653	4	0.056191	0.0931	3	0.09351	0.0047	20

Predictor Screening

	ED_Overflow_Ratio			MDTF_Overflow_Total			MDTF_Overflow_Ratio			Target_Overflow			Cumulative_Overflow		
Predictor	Contribution	Portion	Rank	Contribution	Portion	Rank	Contribution	Portion	Rank	Contribution	Portion	Rank	Contribution	Portion	Rank
"% Obs_Direct_to_MDTF"	0.50773	0.0920	4	386205	0.1971	2	0.37429	0.1863	2	6973	0.0224	3	161924	0.0231	9
"%_of_Fires_Focused_on_Enemy_Artillery"	0.03344	0.0061	15	113612	0.0580	3	0.03280	0.0163	8	237784	0.7641	1	333526	0.0475	7
ACE_Fusion_Expertise	0.04754	0.0086	7	9948	0.0051	19	0.00893	0.0044	20	2589	0.0083	11	67385	0.0096	15
ACE_Fusion_Personnel	0.02772	0.0050	19	19851	0.0101	10	0.01716	0.0085	10	1609	0.0052	18	51132	0.0073	17
ACE_Fusion_Technology	0.03085	0.0056	16	18011	0.0092	13	0.01133	0.0056	17	3868	0.0124	7	52275	0.0074	16
Average_Launch_and_Transit_Time	0.04074	0.0074	10	20363	0.0104	9	0.01435	0.0071	13	2669	0.0086	9	76305	0.0109	13
Crews_Available	0.04614	0.0084	9	18169	0.0093	12	0.01612	0.0080	11	1792	0.0058	16	131290	0.0187	10
ED_Expertise	0.77378	0.1401	3	17806	0.0091	15	0.01033	0.0051	19	5170	0.0166	4	616996	0.0879	5
ED_Personnel	2.24600	0.4068	1	17819	0.0091	14	0.01201	0.0060	16	2040	0.0066	15	1824247	0.2598	1
ED_Technology	1.14598	0.2075	2	9771	0.0050	20	0.01204	0.0060	15	3706	0.0119	8	1076478	0.1533	2
Frequencies_Per_Satellite	0.12962	0.0235	6	57126	0.0292	6	0.03349	0.0167	7	4860	0.0156	5	476831	0.0679	6
Ground_Control_Stations_Available	0.02919	0.0053	18	19237	0.0098	11	0.02388	0.0119	9	4095	0.0132	6	99630	0.0142	11
Initial_UAS	0.04629	0.0084	8	11841	0.0060	17	0.01363	0.0068	14	1630	0.0052	17	41659	0.0059	20
MDTF_Allocation	0.03954	0.0072	13	964596	0.4923	1	1.16558	0.5802	1	2322	0.0075	12	666590	0.0949	4
MDTF_PEF_Expertise	0.02384	0.0043	20	68588	0.0350	5	0.06605	0.0329	4	2298	0.0074	13	80988	0.0115	12
MDTF_PEF_Technology	0.03619	0.0066	14	46713	0.0238	8	0.03753	0.0187	6	1206	0.0039	20	45103	0.0064	19
PE_Expertise	0.04059	0.0074	11	13965	0.0071	16	0.01595	0.0079	12	2045	0.0066	14	69739	0.0099	14
PE_Technology	0.03019	0.0055	17	10013	0.0051	18	0.01082	0.0054	18	1401	0.0045	19	47384	0.0067	18
PED_Personnel_Total	0.04043	0.0073	12	54278	0.0277	7	0.08290	0.0413	3	2666	0.0086	10	209661	0.0299	8
Satellites_Available	0.20593	0.0373	5	81315	0.0415	4	0.04975	0.0248	5	20463	0.0658	2	892503	0.1271	3

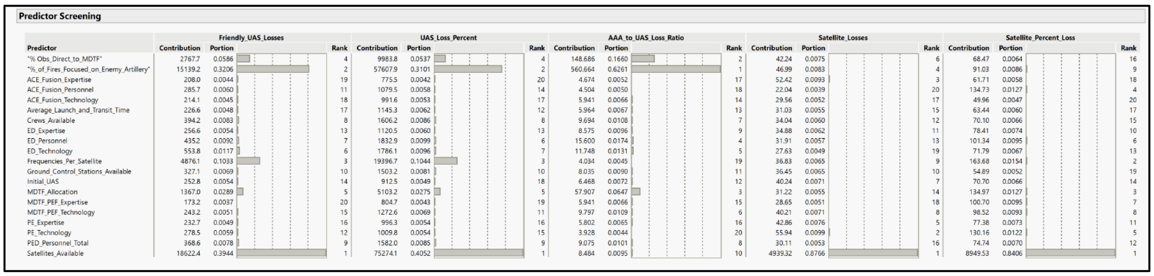
Figure 153: Input Predictor Screening for PED MOP/MOE

Obviously, different outputs have differences in the most influential factors, but the trends are easy to identify. It is clear the '% Obs Direct to MDTF,' MDTF Allocation, PED Personnel Total, Satellites Available, '% of Fires Focused on Enemy Artillery, MDTF Expertise, and MDTF Technology are the most influential for all of the MOP/MOE. (Recall that number of HIMARS was a fixed number for a single MDTF BDE, otherwise it too would likely be a major factor.)



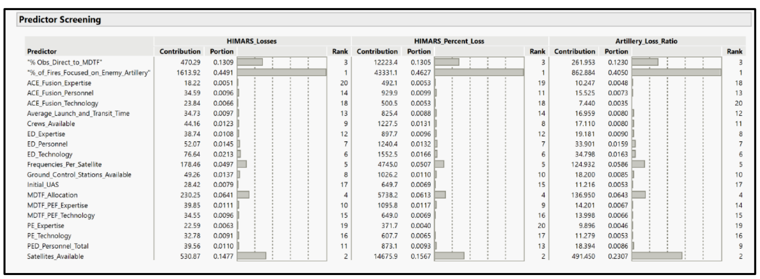
Predictor Screening

Predictor	Total_Loss_Ratio Contribution	Portion	Rank	Enemy_AAA_Losses Contribution	Portion	Rank	Enemy_AAA_Percent_Loss Contribution	Portion	Rank	Enemy_Artillery_Losses Contribution	Portion	Rank	Enemy_Artillery_Percent_Loss Contribution	Portion	Rank
"% Obs_Direct_to_MDTF"	123.570	0.2966	1	36423	0.1325	3	62911	0.1288	3	23293.2	0.0886	3	40528	0.0875	3
"%_of_Fires_Focused_on_Enemy_Artillery"	35.342	0.0848	4	109303	0.3976	1	194100	0.3973	1	96108.9	0.3656	1	155432	0.3356	1
ACE_Fusion_Expertise	2.172	0.0052	17	1078	0.0039	19	2161	0.0044	18	1005.0	0.0038	19	1842	0.0040	20
ACE_Fusion_Personnel	2.952	0.0071	14	1417	0.0052	15	2370	0.0049	17	1311.3	0.0050	15	2341	0.0051	16
ACE_Fusion_Technology	2.125	0.0051	18	1678	0.0061	12	3768	0.0077	10	1354.6	0.0052	14	2055	0.0044	19
Average_Launch_and_Transit_Time	2.389	0.0057	15	1061	0.0039	20	2678	0.0055	14	1018.7	0.0039	18	2490	0.0054	14
Crews_Available	3.775	0.0091	11	1519	0.0055	14	2474	0.0051	15	1628.8	0.0062	11	2835	0.0061	12
ED_Expertise	7.121	0.0171	8	2893	0.0105	8	5194	0.0106	7	2715.6	0.0103	8	4643	0.0100	8
ED_Personnel	13.043	0.0313	6	4302	0.0157	6	8538	0.0175	6	4209.7	0.0160	6	8209	0.0177	6
ED_Technology	9.960	0.0239	7	2204	0.0080	10	4122	0.0084	9	3398.5	0.0129	7	7243	0.0156	7
Frequencies_Per_Satellite	18.746	0.0450	5	14567	0.0530	4	25892	0.0530	4	20266.3	0.0771	4	34715	0.0749	4
Ground_Control_Stations_Available	2.197	0.0053	16	1548	0.0056	13	2987	0.0061	13	1653.8	0.0063	10	3003	0.0065	11
Initial_UAS	1.874	0.0045	20	1384	0.0050	17	2152	0.0044	19	979.3	0.0037	20	2070	0.0045	18
MDTF_Allocation	62.408	0.1498	3	14270	0.0519	5	21800	0.0446	5	15894.4	0.0605	5	33021	0.0713	5
MDTF_PEF_Expertise	2.998	0.0072	13	1410	0.0051	16	2132	0.0044	20	1382.0	0.0053	13	2475	0.0053	15
MDTF_PEF_Technology	6.561	0.0158	9	3212	0.0117	7	4963	0.0102	8	1683.8	0.0064	9	3907	0.0084	9
PE_Expertise	3.246	0.0078	12	2316	0.0084	9	3125	0.0064	12	1499.5	0.0057	12	3580	0.0077	10
PE_Technology	1.960	0.0047	19	1097	0.0040	18	2373	0.0049	16	1201.0	0.0046	17	2276	0.0049	17
PED_Personnel_Total	4.580	0.0110	10	2114	0.0077	11	3388	0.0069	11	1215.7	0.0046	16	2612	0.0056	13
Satellites_Available	109.546	0.2630	2	71082	0.2586	2	131436	0.2690	2	81084.0	0.3084	2	147913	0.3193	2



Predictor Screening

Predictor	Friendly_UAS_Losses Contribution	Portion	Rank	UAS_Loss_Percent Contribution	Portion	Rank	AAA_to_UAS_Loss_Ratio Contribution	Portion	Rank	Satellite_Losses Contribution	Portion	Rank	Satellite_Percent_Loss Contribution	Portion	Rank
"% Obs_Direct_to_MDTF"	2767.7	0.0586	4	9983.8	0.0537	4	148.686	0.1660	2	42.24	0.0075	6	68.47	0.0064	16
"%_of_Fires_Focused_on_Enemy_Artillery"	15139.2	0.3206	2	57607.9	0.3101	2	560.664	0.6261	1	46.99	0.0083	4	91.03	0.0086	9
ACE_Fusion_Expertise	208.0	0.0044	19	775.5	0.0042	20	4.674	0.0052	17	52.42	0.0093	3	61.71	0.0058	18
ACE_Fusion_Personnel	285.7	0.0060	11	1079.5	0.0058	14	4.504	0.0050	18	22.04	0.0039	20	134.73	0.0127	4
ACE_Fusion_Technology	214.1	0.0045	18	991.6	0.0053	17	5.941	0.0066	14	29.56	0.0052	17	49.96	0.0047	20
Average_Launch_and_Transit_Time	226.6	0.0048	17	1145.3	0.0062	12	5.964	0.0067	13	31.03	0.0055	15	63.44	0.0060	17
Crews_Available	394.2	0.0083	8	1606.2	0.0086	8	9.694	0.0108	7	34.04	0.0060	12	70.10	0.0066	15
ED_Expertise	256.6	0.0054	13	1120.5	0.0060	13	8.575	0.0096	9	34.88	0.0062	11	78.41	0.0074	10
ED_Personnel	435.2	0.0092	7	1832.9	0.0099	6	15.600	0.0174	4	31.91	0.0057	13	101.34	0.0095	6
ED_Technology	553.8	0.0117	6	1786.1	0.0096	7	11.748	0.0131	5	27.63	0.0049	19	71.79	0.0067	13
Frequencies_Per_Satellite	4876.1	0.1033	3	19396.7	0.1044	3	4.034	0.0045	19	36.83	0.0065	9	163.68	0.0154	2
Ground_Control_Stations_Available	327.1	0.0069	10	1503.2	0.0081	10	8.035	0.0090	11	36.45	0.0065	10	54.89	0.0052	19
Initial_UAS	252.8	0.0054	14	912.5	0.0049	18	6.468	0.0072	12	40.24	0.0071	7	70.70	0.0066	14
MDTF_Allocation	1367.0	0.0289	5	5103.2	0.0275	5	57.907	0.0647	3	31.22	0.0055	14	134.97	0.0127	3
MDTF_PEF_Expertise	173.2	0.0037	20	804.7	0.0043	19	5.941	0.0066	15	28.65	0.0051	18	100.70	0.0095	7
MDTF_PEF_Technology	243.2	0.0051	15	1272.6	0.0069	11	9.797	0.0109	6	40.21	0.0071	8	98.52	0.0093	8
PE_Expertise	232.7	0.0049	16	996.3	0.0054	16	5.802	0.0065	16	42.86	0.0076	5	77.38	0.0073	11
PE_Technology	278.5	0.0059	12	1009.8	0.0054	15	3.928	0.0044	20	55.94	0.0099	2	130.16	0.0122	5
PED_Personnel_Total	368.6	0.0078	9	1582.0	0.0085	9	9.075	0.0101	8	30.11	0.0053	16	74.74	0.0070	12
Satellites_Available	18622.4	0.3944	1	75274.1	0.4052	1	8.484	0.0095	10	4939.32	0.8766	1	8949.53	0.8406	1



Predictor Screening

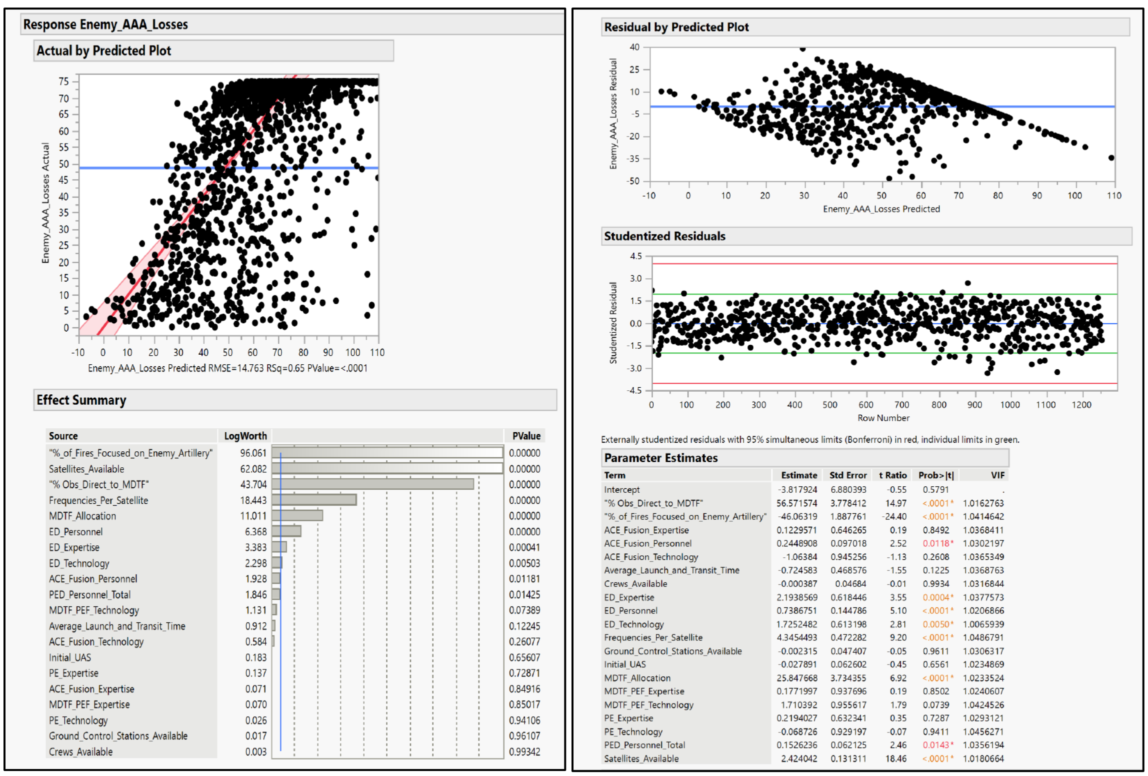
Predictor	HIMARS_Losses Contribution	Portion	Rank	HIMARS_Percent_Loss Contribution	Portion	Rank	Artillery_Loss_Ratio Contribution	Portion	Rank
"% Obs_Direct_to_MDTF"	470.29	0.1309	3	12223.4	0.1305	3	261.953	0.1230	3
"%_of_Fires_Focused_on_Enemy_Artillery"	1613.92	0.4491	1	43331.1	0.4627	1	862.884	0.4050	1
ACE_Fusion_Expertise	18.22	0.0051	20	492.1	0.0053	19	10.247	0.0048	18
ACE_Fusion_Personnel	34.59	0.0096	14	929.9	0.0099	11	15.525	0.0073	13
ACE_Fusion_Technology	23.84	0.0066	18	500.5	0.0053	18	7.440	0.0035	20
Average_Launch_and_Transit_Time	34.73	0.0097	13	825.4	0.0088	14	16.959	0.0080	12
Crews_Available	44.16	0.0123	9	1227.5	0.0131	8	17.110	0.0080	11
ED_Expertise	38.74	0.0108	12	897.7	0.0096	12	19.181	0.0090	8
ED_Personnel	52.07	0.0145	7	1240.4	0.0132	7	33.901	0.0159	7
ED_Technology	76.64	0.0213	6	1552.5	0.0166	6	34.798	0.0163	6
Frequencies_Per_Satellite	178.46	0.0497	5	4745.0	0.0507	5	124.932	0.0586	5
Ground_Control_Stations_Available	49.26	0.0137	8	1026.2	0.0110	10	18.200	0.0085	10
Initial_UAS	28.42	0.0079	17	649.7	0.0069	15	11.216	0.0053	17
MDTF_Allocation	230.25	0.0641	4	5738.2	0.0613	4	136.950	0.0643	4
MDTF_PEF_Expertise	39.85	0.0111	10	1095.8	0.0117	9	14.201	0.0067	14
MDTF_PEF_Technology	34.55	0.0096	15	649.0	0.0069	16	13.998	0.0066	15
PE_Expertise	22.59	0.0063	19	371.7	0.0040	20	9.896	0.0046	19
PE_Technology	32.78	0.0091	16	607.7	0.0065	17	11.279	0.0053	16
PED_Personnel_Total	39.56	0.0110	11	873.1	0.0093	13	18.394	0.0086	9
Satellites_Available	530.87	0.1477	2	14675.9	0.1567	2	491.450	0.2307	2

Figure 154: Input Predictor Screening Combat MOP/MOE

6.3.2.3 Developing Surrogate Models

Armed with the insights from the scatter plots and predictor screenings, surrogate models can be generated using JMP with selected inputs to desired outputs. Given the stochasticity of the enemy fires, it is unlikely that the most basic surrogate modeling paradigm will sufficiently model the SoS, but they do provide additional verification of the most influential factors. Figure 155 shows basic fit model using least squares. The fit of the surrogate model is poor, but the information obtains is important. The Effect Summary shows the most important input variables (highlighted in blue with p-value less than 0.05%)

Additionally, under the Parameter Estimates, the VIF (variance inflation factors) measure the multicollinearity of the variables. Any value more than 5 is associated with an 80% correlation between predictors, however, all our input values are approximately 1. High correlation creates wrong results and unstable variances.

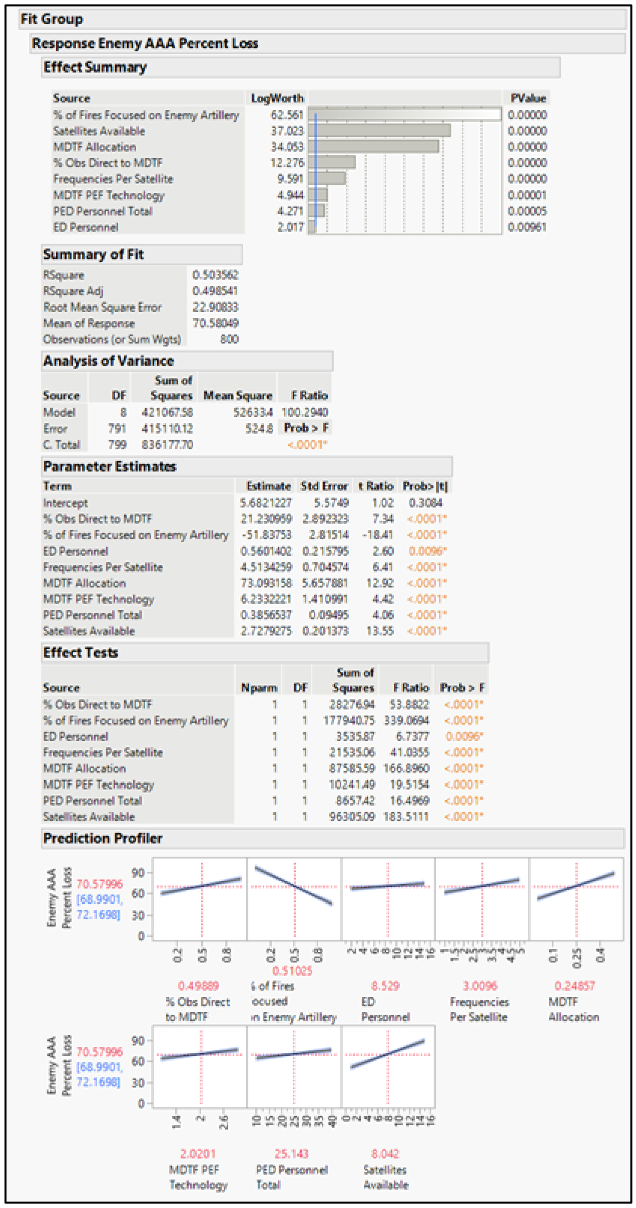


Source	LogWorth	PValue
"%_of_Fires_Focused_on_Enemy_Artillery"	96.061	0.00000
Satellites_Available	62.082	0.00000
"% Obs_Direct_to_MDTF"	43.704	0.00000
Frequencies_Per_Satellite	18.443	0.00000
MDTF_Allocation	11.011	0.00000
ED_Personnel	6.368	0.00000
ED_Expertise	3.383	0.00041
ED_Technology	2.298	0.00503
ACE_Fusion_Personnel	1.928	0.01181
PED_Personnel_Total	1.846	0.01425
MDTF_PEF_Technology	1.131	0.07389
Average_Launch_and_Transit_Time	0.912	0.12245
ACE_Fusion_Technology	0.584	0.26077
Initial_UAS	0.183	0.65607
PE_Expertise	0.137	0.72871
ACE_Fusion_Expertise	0.071	0.84916
MDTF_PEF_Expertise	0.070	0.85017
PE_Technology	0.026	0.94106
Ground_Control_Stations_Available	0.017	0.96107
Crews_Available	0.003	0.99342

Term	Estimate	Std Error	t Ratio	Prob>|t|	VIF
Intercept	-3.817924	6.880393	-0.55	0.5791	.
"% Obs_Direct_to_MDTF"	56.571574	3.778412	14.97	<.0001*	1.0162763
"%_of_Fires_Focused_on_Enemy_Artillery"	-46.06319	1.887761	-24.40	<.0001*	1.0414642
ACE_Fusion_Expertise	0.1229571	0.646265	0.19	0.8492	1.0368411
ACE_Fusion_Personnel	0.2448908	0.097018	2.52	0.0118*	1.0302197
ACE_Fusion_Technology	-1.06384	0.945256	-1.13	0.2608	1.0365349
Average_Launch_and_Transit_Time	-0.724583	0.468576	-1.55	0.1225	1.0368763
Crews_Available	-0.000387	0.04684	-0.01	0.9934	1.0316844
ED_Expertise	2.1938569	0.618446	3.55	0.0004*	1.0377573
ED_Personnel	0.7386751	0.144786	5.10	<.0001*	1.0206866
ED_Technology	1.7252482	0.613198	2.81	0.0050*	1.0065939
Frequencies_Per_Satellite	4.3454493	0.472282	9.20	<.0001*	1.0486791
Ground_Control_Stations_Available	-0.002315	0.047407	-0.05	0.9611	1.0306317
Initial_UAS	-0.027891	0.062602	-0.45	0.6561	1.0234869
MDTF_Allocation	25.847668	3.734355	6.92	<.0001*	1.0233524
MDTF_PEF_Expertise	0.1771997	0.937696	0.19	0.8502	1.0240607
MDTF_PEF_Technology	1.710392	0.955617	1.79	0.0739	1.0424526
PE_Expertise	0.2194027	0.632341	0.35	0.7287	1.0293121
PE_Technology	-0.068726	0.929197	-0.07	0.9411	1.0456271
PED_Personnel_Total	0.1526236	0.062125	2.46	0.0143*	1.0356194
Satellites_Available	2.424042	0.131311	18.46	<.0001*	1.0180664

Figure 155: Results of Method of Least Squares Basic Fit Model

As an alternative option, stepwise regression forward was conducted by slowly adding variables to see which variable have most correlation/influence on the response. While also not a good fit, it also shows the effects of the variables that can be used in other analysis. The two failed surrogates combined with the previous scatterplots and prediction profilers can be used to eliminate unnecessary variables and reduce complexity.



Fit Group

Response Enemy AAA Percent Loss

Effect Summary

Source	LogWorth	PValue
% of Fires Focused on Enemy Artillery	62.561	0.00000
Satellites Available	37.023	0.00000
MDTF Allocation	34.053	0.00000
% Obs Direct to MDTF	12.276	0.00000
Frequencies Per Satellite	9.591	0.00000
MDTF PEF Technology	4.944	0.00001
PED Personnel Total	4.271	0.00005
ED Personnel	2.017	0.00961

Summary of Fit

RSquare	0.503562
RSquare Adj	0.498541
Root Mean Square Error	22.90833
Mean of Response	70.58049
Observations (or Sum Wgts)	800

Analysis of Variance

Source	DF	Sum of Squares	Mean Square	F Ratio
Model	8	421067.58	52633.4	100.2940
Error	791	415110.12	524.8	Prob > F
C. Total	799	836177.70		<.0001*

Parameter Estimates

Term	Estimate	Std Error	t Ratio	Prob>|t|
Intercept	5.6821227	5.5749	1.02	0.3084
% Obs Direct to MDTF	21.230959	2.892323	7.34	<.0001*
% of Fires Focused on Enemy Artillery	-51.83753	2.81514	-18.41	<.0001*
ED Personnel	0.5601402	0.215795	2.60	0.0096*
Frequencies Per Satellite	4.5134259	0.704574	6.41	<.0001*
MDTF Allocation	73.093158	5.657881	12.92	<.0001*
MDTF PEF Technology	6.2332221	1.410991	4.42	<.0001*
PED Personnel Total	0.3856537	0.09495	4.06	<.0001*
Satellites Available	2.7279275	0.201373	13.55	<.0001*

Effect Tests

Source	Nparm	DF	Sum of Squares	F Ratio	Prob > F
% Obs Direct to MDTF	1	1	28276.94	53.8822	<.0001*
% of Fires Focused on Enemy Artillery	1	1	177940.75	339.0694	<.0001*
ED Personnel	1	1	3535.87	6.7377	0.0096*
Frequencies Per Satellite	1	1	21535.06	41.0355	<.0001*
MDTF Allocation	1	1	87585.59	166.8960	<.0001*
MDTF PEF Technology	1	1	10241.49	19.5154	<.0001*
PED Personnel Total	1	1	8657.42	16.4969	<.0001*
Satellites Available	1	1	96305.09	183.5111	<.0001*

Figure 156: Stepwise Regression Example (Enemy AAA Percent Loss)

Such regression fits were completed for a variety of output variables to confirm the most influential factors. The following input variables were selected:

AISR PED D3A Surrogate Model Input Parameters									
PED					HIMARS				
0.0	<=	PE Expertise	<=	10.0	0.0	<=	% Fires Focused on Enemy Artillery	<=	1.0
1.0	<=	PE Technology	<=	3.0	UAS				
0.0	<=	MDTF Expertise	<=	10.0	0	<=	Total UAS in Theater	<=	75
1.0	<=	MDTF Technology	<=	3.0	2	<=	Average Launch and Transit Time	<=	6
0.0	<=	% Obs Direct to MDTF	<=	0.5	1	<=	Initial UAS	<=	50
0.0	<=	MDTF Allocation	<=	0.5	0	<=	Total Crews	<=	50
0.0	<=	PED Personnel Total	<=	50.0	0	<=	Ground Control Stations Available	<=	50
0.0	<=	ED Expertise	<=	10.0	Satellites				
1.0	<=	ED Technology	<=	3.0	1	<=	Satellites Available	<=	15
0.0	<=	ED Personnel	<=	30.0	1	<=	Frequencies Per Satellite	<=	10
Fusion									
1	<=	ACE Fusion Expertise	<=	10					
1	<=	ACE Fusion Technology	<=	3					
1	<=	ACE Fusion Personnel	<=	30					

Figure 157: Input Variables for Surrogate Modeling

Several methods to properly fit surrogate models were attempted, including Random Forests, Stepwise Regressions, and Response Surface Methods. However, none were able to provide adequate goodness of fit. Therefore, built in artificial neural nets were used to create the surrogate models. Ultimately, neural net surrogate models were successfully created for key MOP/MOE: Cumulative Overflow, Enemy AAA Percent Loss, Enemy AAA to UAS Loss Ratio, Artillery Loss Ratio, Total Loss Ratio, UAS Loss Percent, HIMARS Loss Percent. Surrogates could not be made for the Satellite Percent Loss due the nature of the assumed

Each model was generated using 1500 designated training point then validated against an additional 500 validations points (orange), and final confirmed against a different set of 500 test points (green). From the 1500 training points, 33% were held back as initial validation. The resulting equation was then exported into JMP graph builder to test the additional 500 validation and 500 test points (see Figure 159 and Figure 160).

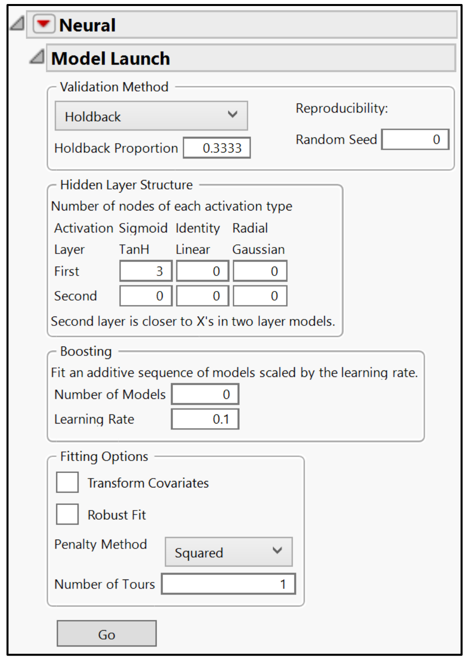


Figure 158: Neural Net Settings Panel

The number of nodes at each of the two layers were varied as were the associate activation functions associated with those nodes to try and maximize R^2 values (coefficients of determination) as close to 1 as a possible for both initial test and holdback validation points. All the neural net created surrogate models were able to achieve a minimum R^2 value of 0.95 for training, validation, and test values. The goodness of fit was evaluated by the actual versus predicted plots and the residuals vs predicted plots as well the predicted model fit error.

For the actual versus predicted plots, goodness of fit is assessed using the 95% confidence lines. Ideally the 95% confidence lines should be as close to the fit line as possible. Points that lie outside of the confidence line are failed points. Every model had clusters of points that failed. However, they were not always the same points in the design space. While the 95% confidence lines were farther apart from the fit line than preferred, they ran generally parallel and were not divergent. Most points fell within these confidence lines. Creating surrogate predictive models for nonphysical system without closed form solutions and large amounts of stochasticity is extremely difficult. For these types of problems and for our intended use, the current fits are sufficient. For models that could not obtain suitable R^2 or goodness of fit, surrogate models were made on the natural logarithmic transforms of those output variables (see Figure 160).

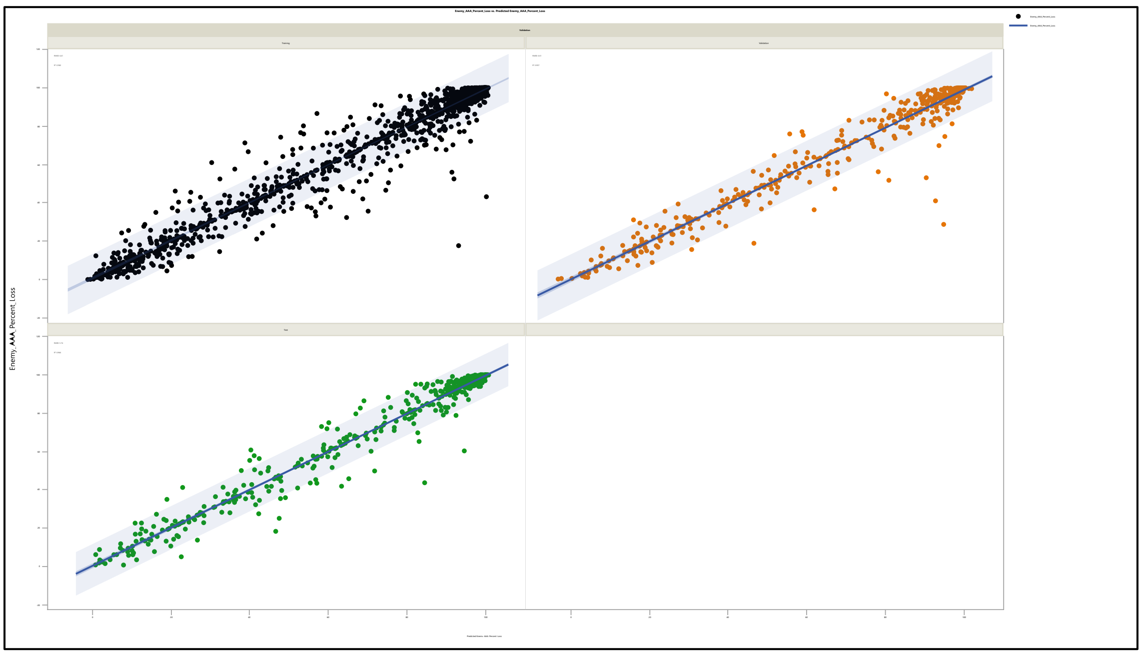


Figure 159: Enemy AAA Percent Loss Actual vs. Predicted

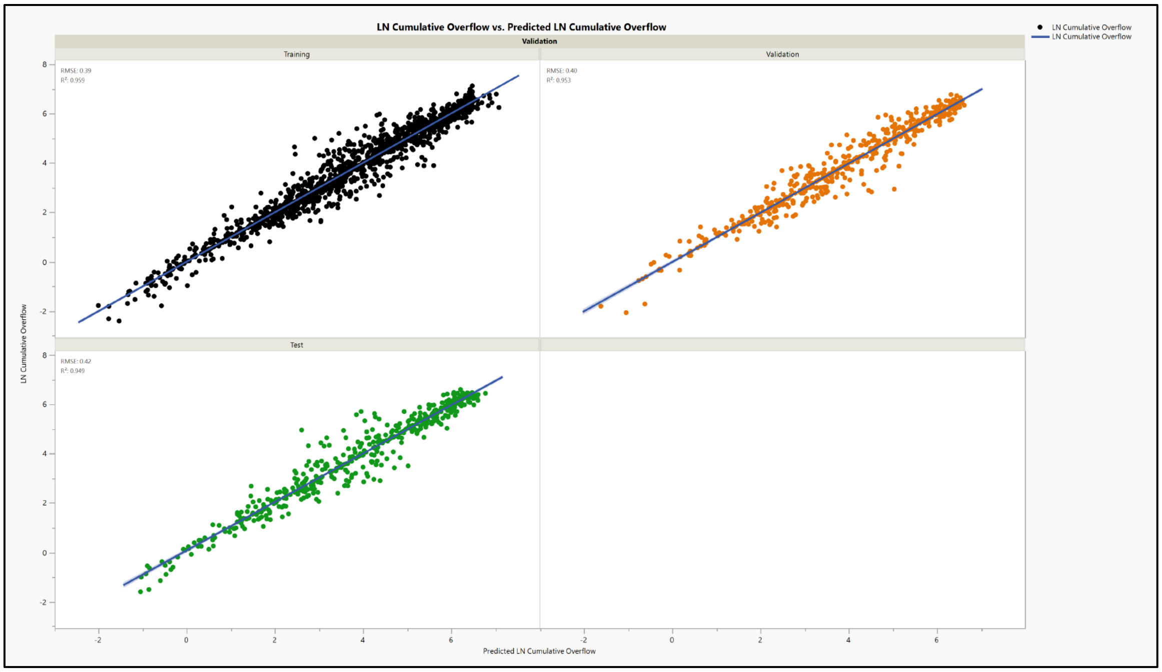


Figure 160: Cumulative Overflow (LN Transform) Actual Vs. Predicted

6.3.3 Optimization of Inputs Using Surrogate Models

For a simulation paradigm like SD, one may wonder why surrogate models should be made at all. Typically, surrogate models are used to replace computational inexpensive simulation by creating instantly executable mathematical relationships between input and output variables of interest for statistical exploration. For something as computationally inexpensive as SD, the creation of surrogate models may seem unnecessary. However, because of the dimensionality of the executable architecture, the surrogate models and statistical analysis help establish ideal values for the most desirable outcomes. These can serve as benchmarks for acquisition efforts or as start points for model exploration. The predictive profiler was created for the surrogate models (see Figure 161). The rows show outputs and the columns represent the inputs; each has a specified numeric range. The black lines indicate the relationship between the intersecting inputs and outputs.

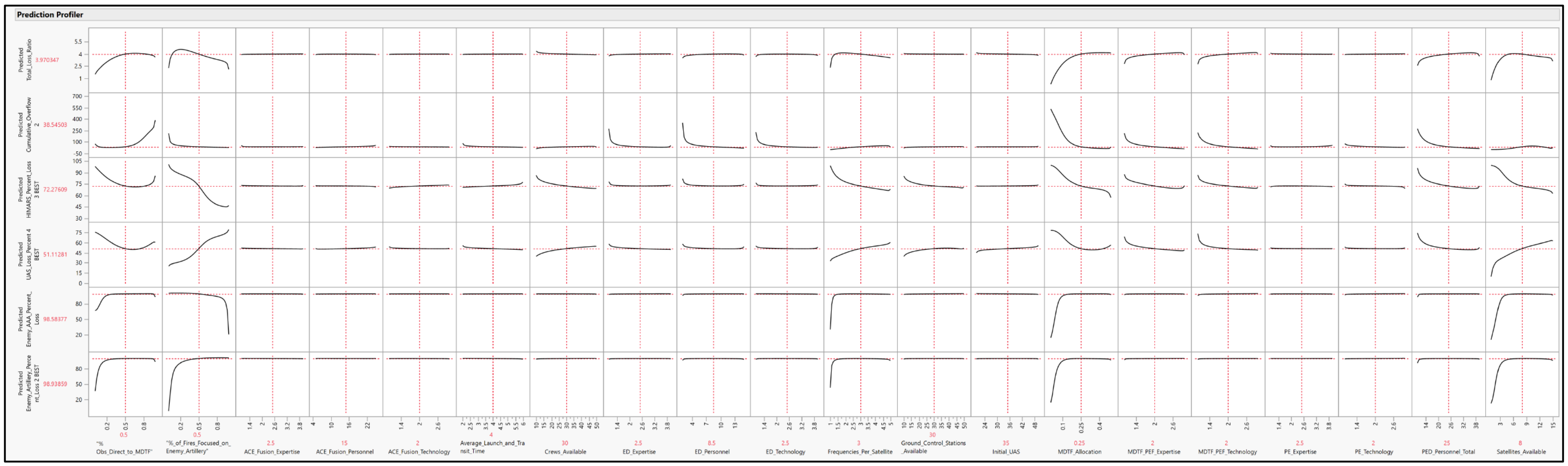


Figure 161: Surrogate Model Prediction Profiler

Flat lines indicate that there is little or no change in an output variable regardless of the value of the input variable. The curved lines indicated dependency. Note that the input variables with the generally flat lines are the same inputs predicted in previous analysis.

With the prediction profiler we can optimize the input values by maximizing the desirability of the outputs. In Figure 161 has 8 outputs, some of which may appear redundant. For example, the loss ratio of friendly to enemy assets itself accounts for the ratio of enemy to friendly loss percentages. However, stakeholders may want to maximize the loss ratio but also minimize friendly loss percent which may not be the same priority. For the optimization settings, it is imperative to prioritize objective functions. This requires not only interviews with stakeholders or SMEs to determine priorities, but also for the end user to understand the effects of the relationships on the optimization.

Two examples are provided to demonstrate this point. For the first analysis, the following outputs selected in order of importance from highest to lowest: Total Loss Ratio (maximize), Cumulative Overflow (minimize), HIMARS Percent Loss (minimize), UAS Percent Loss (minimize). This order means that the end user wants to destroy a higher percentage of enemy than the percentage of friendly destroyed; minimize the amount of information lost during processing; preserve limited assets (HIMARS); and finally preserve 'attritable' assets (UAS). The desirability column is added to the end and desirability values were set to the conditions specified. The optimization then resets the input values to optimal settings (Figure 162).

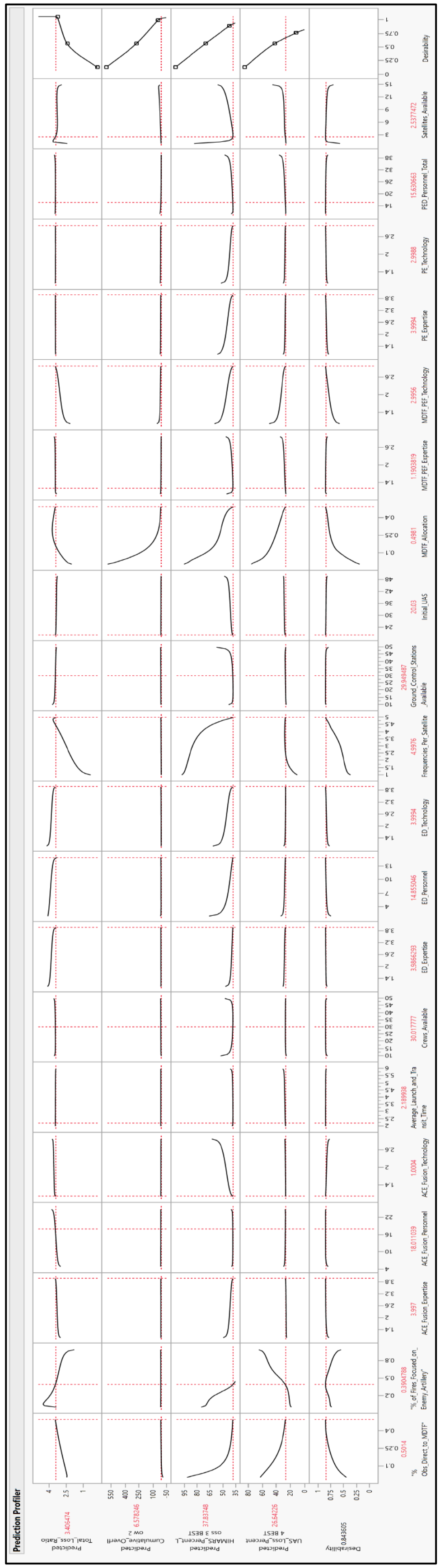


Figure 162: Prediction Profiler with Optimized Desirability Case 1

Table 30: Maximized Desirability Input Values Case 1

Maximized Desirability Settings														
PED						Fusion			HIMARS			UAS		
PE Expertise	=	3.990	MDTF Allocation	=	0.498	ACE Fusion Expertise	=	3.990	% Fires Focused on Enemy Artillery	=	0.390	Total UAS in Theater	=	75.000
PE Technology	=	2.990	PED Personnel Total	=	15.600	ACE Fusion Technology	=	1.000	Satellites			Average Launch and Transit Time	=	2.180
MDTF Expertise	=	1.190	ED Expertise	=	3.990	ACE Fusion Personnel	=	18.000	Satellites Available	=	2.500	Initial UAS	=	20.030
MDTF Technology	=	3.000	ED Technology	=	3.990				Frequencies Per Satellite	=	4.990	Total Crews	=	30.000
% Obs Direct to MDTF	=	0.500	ED Personnel	=	14.850							Ground Control Stations Available	=	29.940

Note that the surrogate models treat all values as continuous so unrealistic decimal values may result. The sliders on the prediction profiler can be slide manually to instantly see the effects on all other variables when rounding to nearest integers or adjusting sliders to attainable, present day values (i.e targetable technology goals). The importance of selecting appropriate MOP and MOE as the objective functions and applying the proper desirability cannot be understated. The optimization algorithm seeks to find the local optimum desirable solution and does not use human logic to understand the implications of the mathematical solution. Thus, it is imperative that the solutions not be taken at face-value and be transferred back into the executable architecture visualization to see the effects over time. Different prioritization and inclusion of output variables will yield very different results.

For comparison case, the following outputs selected in order of importance from highest to lowest: Total Loss Ratio (maximize), Predicted Enemy Artillery Percent Loss (maximize), Predicted Enemy AAA Percent Loss (maximize), HIMARS Percent Loss (minimize), UAS Percent Loss (minimize). Cumulative Overflow (minimize). Like the previous case, this order means that the end user wants to destroy a higher percentage of enemy than the percentage of friendly destroyed but do so by maximizing enemy losses. (Figure 163).

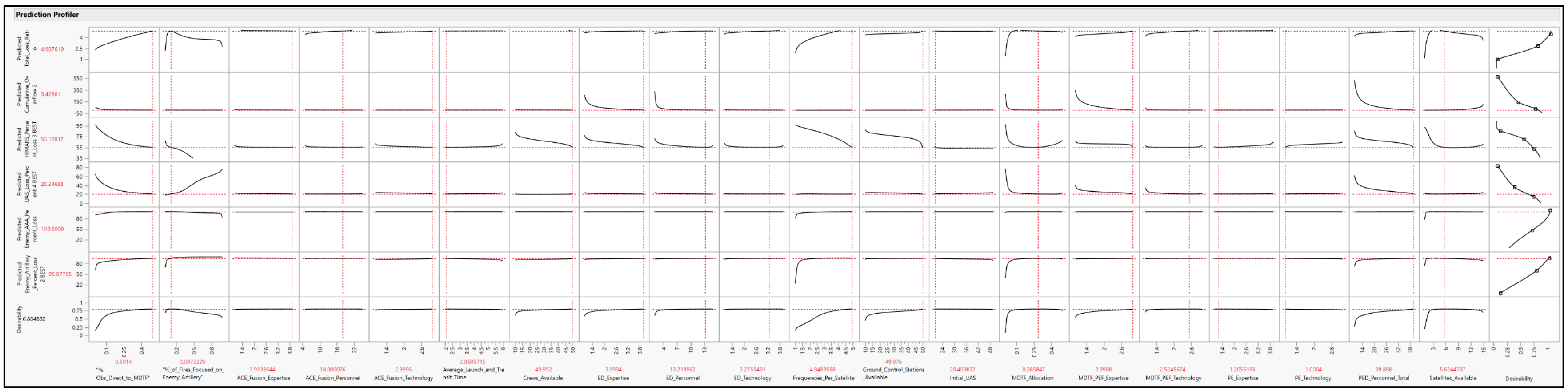


Figure 163: Prediction Profiler with Optimized Desirability Case 2

Table 31: Maximized Desirability Input Values Case 2

Maximized Desirability Settings														
PED						Fusion			HIMARS			UAS		
PE Expertise	=	1.205	MDTF Allocation	=	0.285	ACE Fusion Expertise	=	3.910	% Fires Focused on Enemy Artillery	=	0.097	Total UAS in Theater	=	75.000
PE Technology	=	1.000	PED Personnel Total	=	39.898	ACE Fusion Technology	=	2.990	Satellites			Average Launch and Transit Time	=	2.060
MDTF Expertise	=	2.998	ED Expertise	=	3.990	ACE Fusion Personnel	=	18.000	Satellites Available	=	5.624	Initial UAS	=	20.409
MDTF Technology	=	2.524	ED Technology	=	3.270				Frequencies Per Satellite	=	4.948	Total Crews	=	49.990
% Obs Direct to MDTF	=	0.500	ED Personnel	=	13.218							Ground Control Stations Available	=	49.976

Table 32: Comparison of Desirability Runs shows the recommended outputs of both runs. The values have been rounded to realistic integer as applicable. Those values with the greatest differences between runs have been highlighted in yellow. The ramifications of these differences and the reasons for them is discussed in the next section.

Table 32: Comparison of Desirability Runs

	Desirability Settings Compare		
	Parameter	Case 1	Case 2
PED	PE Expertise	4.0	1.2
	PE Technology	3.0	1.0
	MDTF Expertise	1.2	3.0
	MDTF Technology	3.0	2.5
	% Obs Direct to MDTF	0.5	0.5
	MDTF Allocation	0.5	0.3
	PED Personnel Total	16.0	40.0
	ED Expertise	4.0	4.0
	ED Technology	4.0	3.3
	ED Personnel	15.0	13.0
Fusion	ACE Fusion Expertise	4.0	3.9
	ACE Fusion Technology	1.0	3.0
	ACE Fusion Personnel	18.0	18.0
Fires	% Fires Focused on Enemy Artillery	0.4	0.1
Satellite	Satellites Available	3.0	6.0
	Frequencies Per Satellite	5.0	5.0
UAS	Total UAS in Theater	75.0	75.0
	Average Launch and Transit Time	2.2	2.1
	Initial UAS	20.0	20.0
	Total Crews	30.0	50.0
	Ground Control Stations Available	29.0	50.0

6.4 Adjustments and Visualization

Now that initial desirable settings have been identified, they can be reintroduced into the SD executable architecture to visualize the effects over time and against the random effects of enemy fires (by varying NOISE SEED). The uncertainty from the enemy noise is difficult to capture in surrogate models. The intent is for the output solution to be robust enough that the desired MOE maintain close to the same ratio even if the MOP vary.

The input values from each of the cases were updated in the executable architecture. The MOP/MOE graphs shown side-by-side for comparison for the runs are shown in Figure 164, Figure 165, Figure 166, and Figure 167. The need for proper selection and prioritization of optimization objective functions and the need to visualize the effects over time becomes apparent. The first optimization case prioritized the loss ratio but also sought to reduce the observable overflow as the second priority. The numeric optimization has no understanding of the actual operational environment. In order to maximize the loss ratio and minimize the lost observables, the solution to the optimization of the surrogate models simply minimized the number of UAS and satellites overhead, resulting in fewer friendly destroyed (reducing the denominator).and less intelligence collected. While the settings did meet the objective requirements, it fails to meet actual operational requirements. The second case, while also prioritizing the loss ratio, added the maximization of enemy loss percentages as second priorities to drive up the numerator. The observable overflow was placed as the lowest priority. As a result, more friendly assets were employed, enemy AAA were eliminated, and enemy artillery driven towards zero.

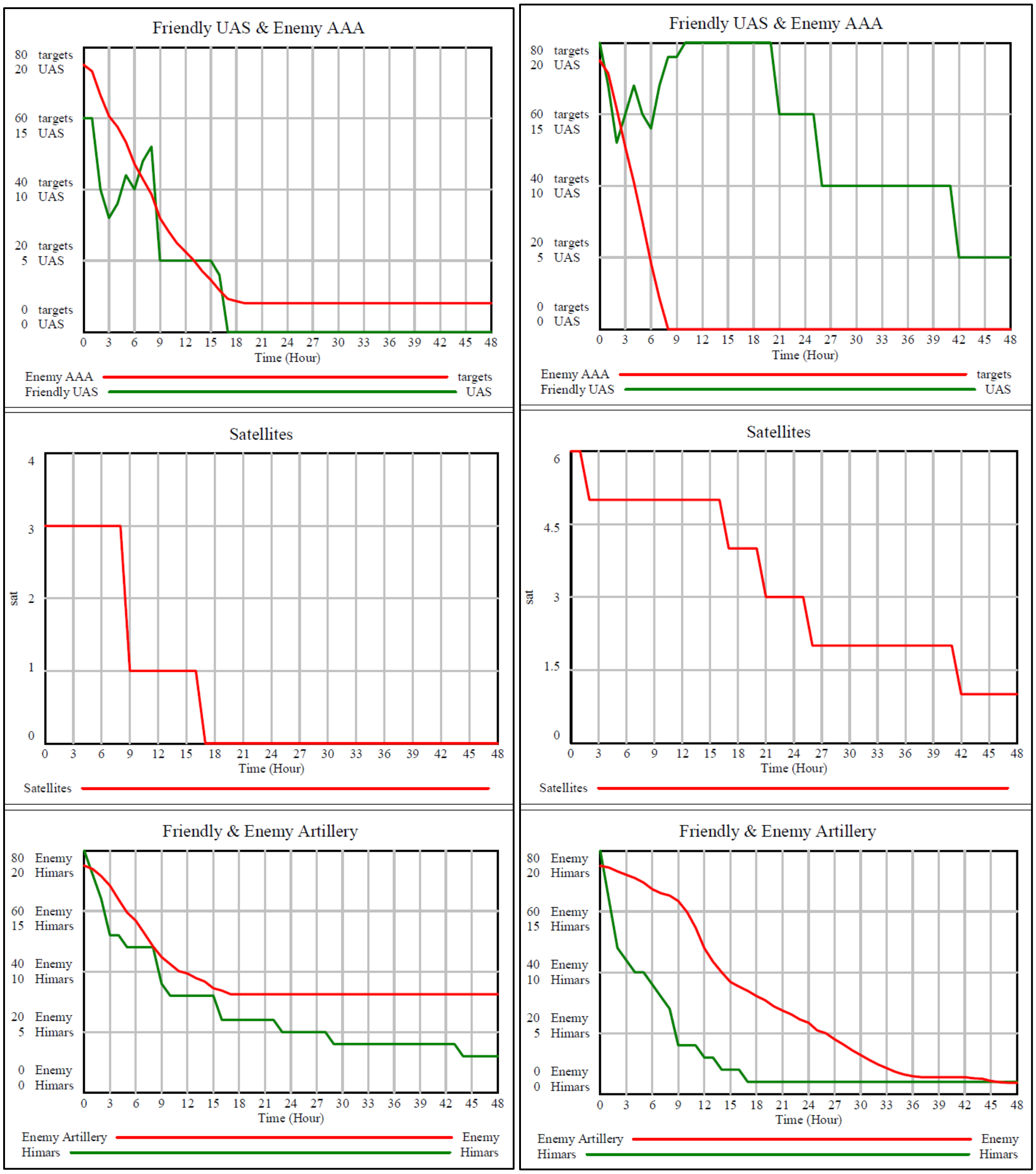


Figure 164: Comparison of Operational MOP/MOE A (Case 1 Left) (Case 2 Right)

Figure 164 shows that the reduced number of satellites from Case 1, result in a complete loss of UAS capacity and as consequently the loss of key observables to target enemy artillery at the 17-hour mark. Despite having only 10% of fires focused on enemy counter battery compared to 40% in Case 1, Case 2 was more effective in destroying enemy artillery, albeit over a longer period.

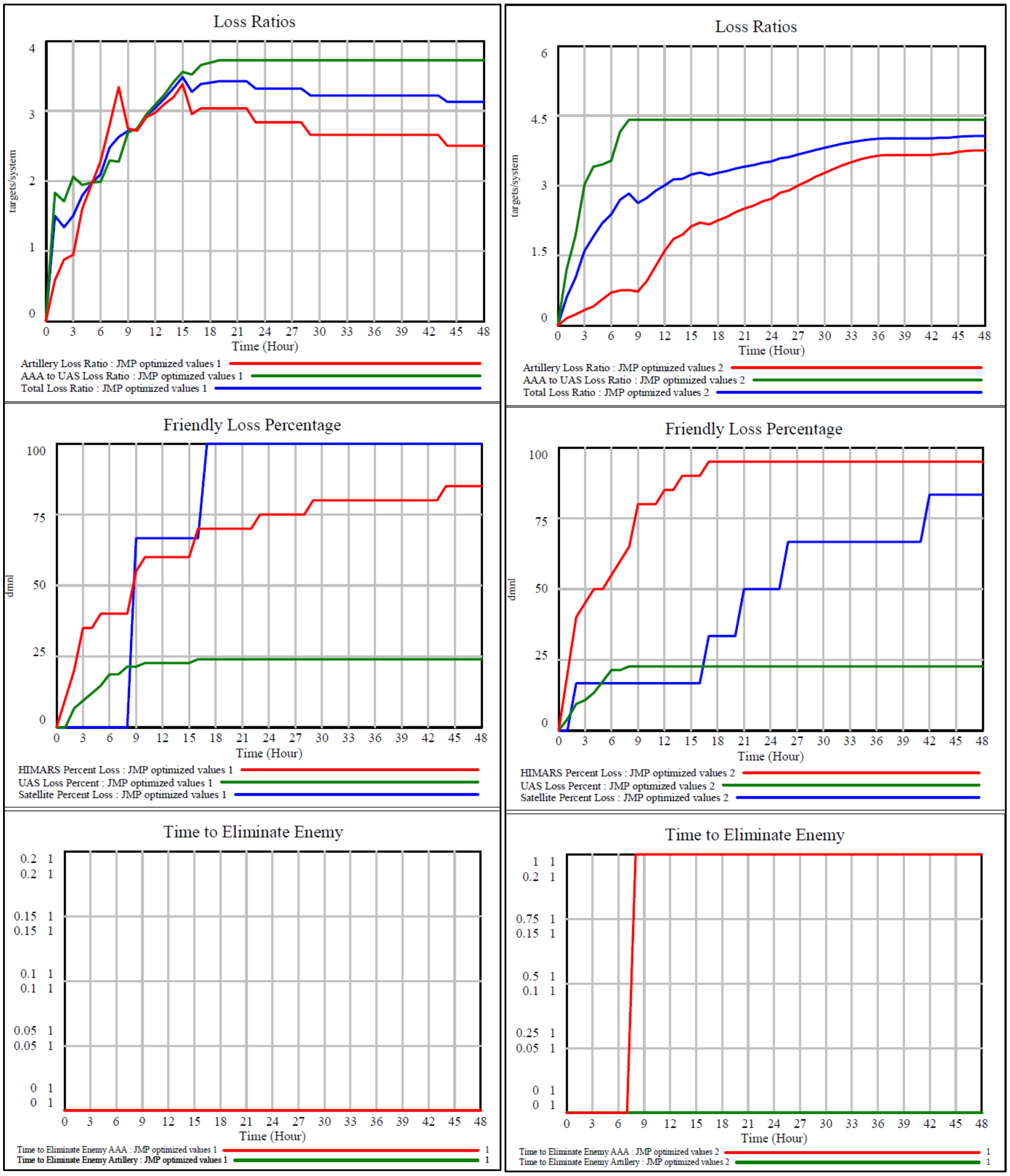


Figure 165: Comparison of Operational MOP/MOE B (Case 1 Left)(Case 2 Right)

Figure 165 shows the benefit of prioritizing the numerator of the loss ratio. Case 2 had a higher loss ratio overall, equal UAS loss percentages, and only slightly higher HIMARS loss percentages.

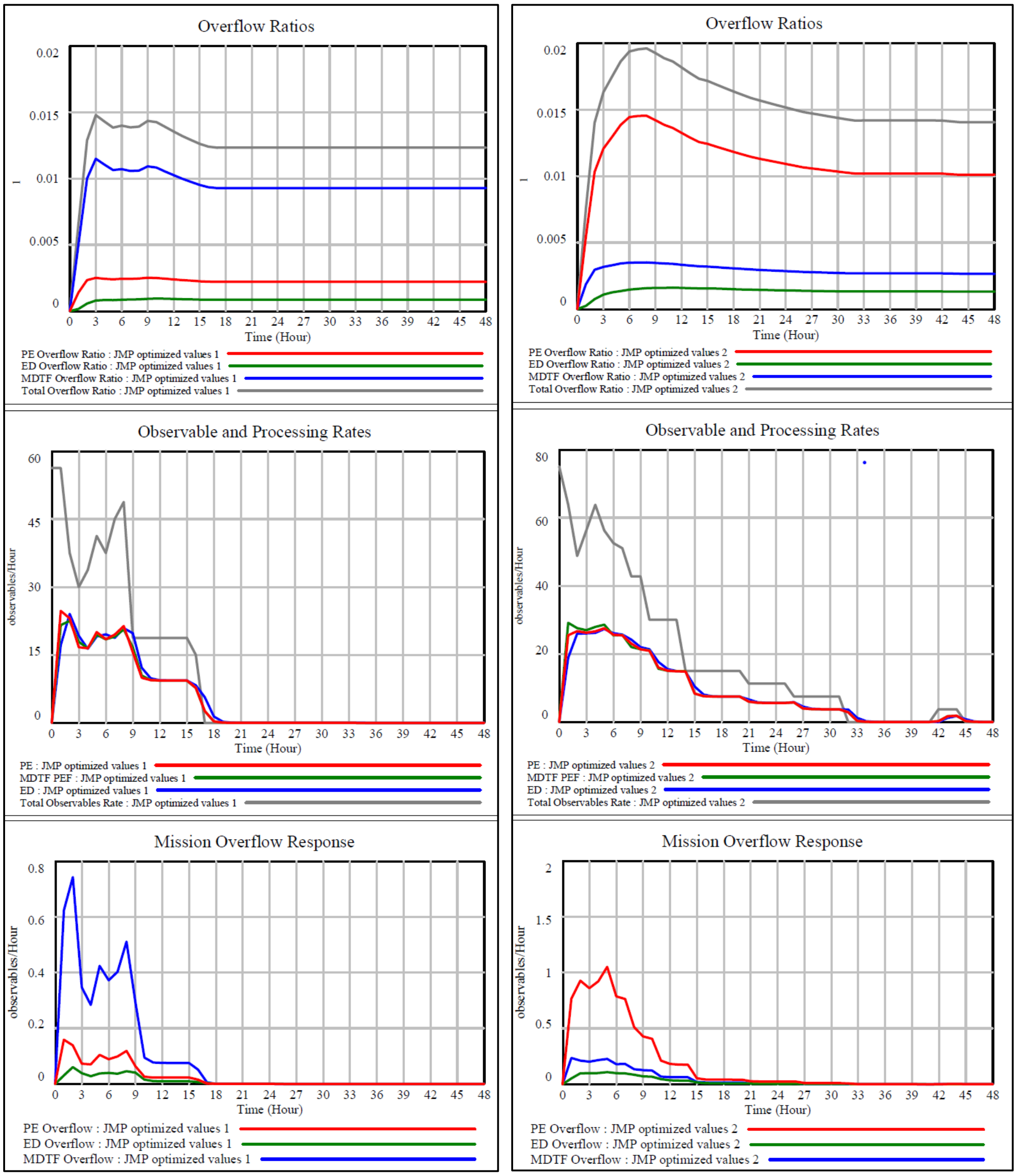


Figure 166: Comparison of Operational MOP/MOE C (Case 1 Left)(Case 2 Right)

Figure 166 shows that Case 2 resulted in only slightly higher observable overflows despite having more assets overhead for longer periods of time with only slightly more capacity utilization and no overwork (Figure 167). This is due to the increased PED personnel required from the optimization.

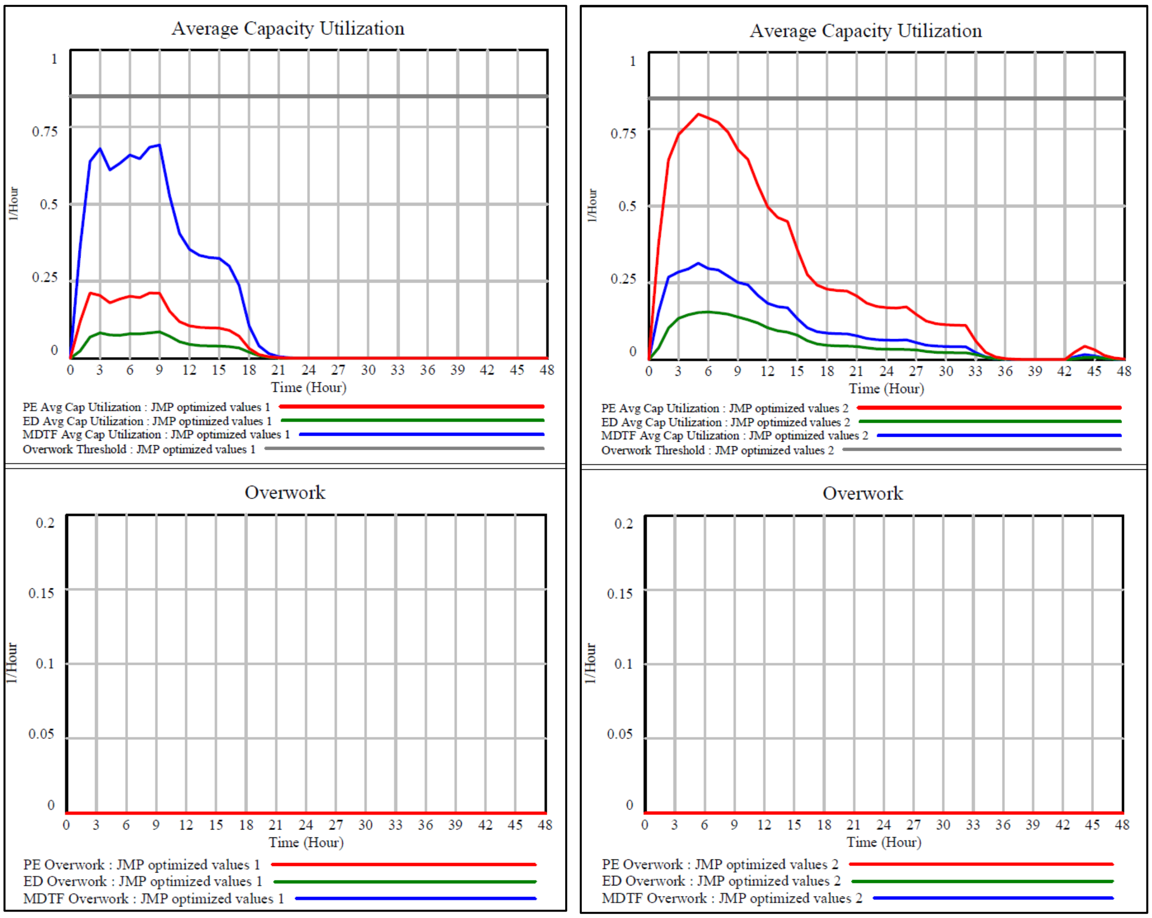


Figure 167: Comparison of Operational MOP/MOE D (Case 1 Left)(Case 2 Right)

These two cases provide numerically generated inputs based on desirability, but they are not absolutes, nor do they guarantee global maximum. The benefit of the executable architecture is that these values can serve as starting points to manual adjustments of values based on current conditions/capabilities and allow immediate visualization of effects or fine tuning to change all outputs.

6.5 Results and Findings for Research Question 3

Research Question 3 sought to demonstrate the ability of the executable architecture could be used to identify elements and values of the larger SoS architecture that have the greatest impact on operational MOP/MOE. Unlike Research Question 2, the larger SoS had many feedback loops requiring a means to compare trades and observe the effects of multiple subsystems that were built in Section 4.3. While Research Question 2 showed the ability to conduct sensitivity analysis and optimization to find ideal settings for the smaller system, the overall SoS executable architecture had far more variables and the inclusion of stochastic enemy actions that induced highly influential noise.

To account for stochasticity in normally deterministic M&S paradigm, the two cases provide numerically generated inputs based on desirability, but they are not absolutes, nor do they guarantee global maximum. A Latin Hypercube DOE was constructed to include the same 2500 points in the design space for 410 noise variables. The averages of the results for each point were used to construct surrogate models using artificial neural nets. Two cases with different priorities and desirability of MOP/MOE yielded drastically different results to demonstrate the importance of proper MOP/MOE desirability selection. The results were input back into the SD executable architecture as start values to visualize and evaluate the effects over time while making univariate changes to adjust value to realistic or feasible goals. If desired, the overall system could be optimized again in the SD executable architecture by provide narrowed ranges around the values output from the optimization of the surrogate models. This type of optimization was demonstrated in RQ2 and was not repeated in the interest of brevity.

Several key takeaways were observed. First, when placed in the larger operational context against operational MOP/MOE without a standardized input the PED manning, technology, and experience levels varied significantly from the values found in RQ2. Most namely, the allocation of observables and the allocation of MDTF PED personnel. This example highlights the necessity to evaluate SoS holistically in the larger operational context when discerning manning, asset, and technology allocations. The SD executable architecture provides a rapid means to do so with multiple stakeholders.

The use of surrogate models and statistical analysis tools allowed for several different means to identify of the most influential variables and the inputs for the executable architecture. By comparing the very different behaviors for two different sets of recommended inputs base on prioritization of MOP/MOE yields insight into how influential the variables selected are. The variables highlighted in yellow in Table 32 indicate values that changed the most between Case 1 and Case 2, but they do not necessary indicate these variables are the most influential of the influential variables. The variables that did not change between Case 1 and Case 2 are just as important but have more certainty in their values. Thus, when returning to the SD executable architecture those values can be fixed as targetable values and only those highlighted in yellow should be undergo univariate change as necessary to obtain the desired behavior.

All the variables used in the cases were identified in Section 6.3. Interestingly, these variables spanned the DOTmLPF-P spectrum (Figure 168: Important Variables Across DOTmLPF-P Figure 168)

Doctrine				Org.	Training				Materiel						Personnel					
% Obs Direct to MDTF	Average Launch and Transit Time	Total UAS in Theater	% Fires Focused on Enemy Artillery	MDTF Allocation	ACE Fusion Expertise	ED Expertise	MDTF Expertise	PE Expertise	Frequencies Per Satellite	ACE Fusion Technology	ED Technology	MDTF Technology	PE Technology	Satellites Available	PED Personnel Total	ED Personnel	ACE Fusion Personnel	Ground Control Stations Available	Total Crews	Initial UAS

Figure 168: Important Variables Across DOTmLPF-P Spectrum

Two major exceptions of note were the number of crews and the number of ground control stations. Because of the way that the SD executable architecture is constructed to reflect real limit on real operations, 5 items limit the number of UAS overhead at any given time: Initial UAS (UAS desired overhead); crew limit, ground control station limit, the frequency limit, and the UAS Pool level (see Section 4.6.1). This constraint is not reflected in the surrogate models, as logic cannot be applied. From Table 32, it can be seen that Case 1 and Case 2 call for a significantly different number of Total Crews (30 versus 50) and Ground Control Stations Available (29 versus 50). However, the frequency limit (satellites multiplied by frequencies per satellite) were much lower (15 versus 35) and were thus the limiting factor. Therefore, any number of crews or ground control stations greater than the most restrictive limit has no effect whatsoever on the behavior of the system. If, however, crew or ground control stations were the limiting constraint, they would have set the maximum for the number of UAS Overhead (at least in initially) and had a significant impact on the behavior of the system over time. This observation highlights the importance of having an executable architecture that is both interactive and is built with SMEs and/or stakeholders to improve their knowledge of the system interactions rather than just rely on a data-in, data-out methodology.

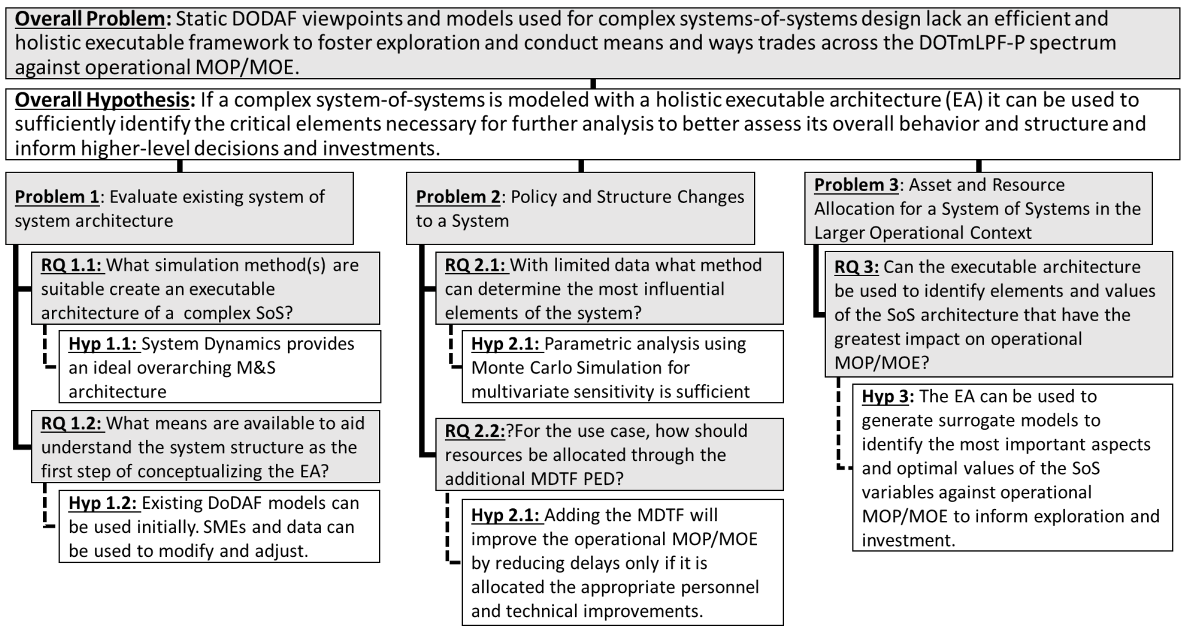
This framework also yielded other interesting observations from comparison of the two cases. To obtain maximized enemy to friendly loss ratios, both cases called for maximum UAS in theater (75) but also both called for only 20 Initial UAS. For Case 1, the UAS Overhead were limited to 15 due to the frequency limit, as discussed. In Case 2, however, the frequency limit was 35. The Initial UAS was the limiting value for UAS Overhead. This is significant because it provides a valuable insight that goes against assumed doctrine. To maximize immediate observables and targeting, one could logically expect to put the maximum amount of UAS overhead, but the effect of capping the UAS overhead allowed a longer use of the reserve pool until satellite attrition limited UAS overhead.

Additionally, comparison of the two cases provided insight into proper targeting prioritization with regards to HIMARs focus. In both cases, less than 40% of fires were directed at enemy counter battery, which means most fires were directed at the enemy AAA. This confirms the strategy of first eliminating enemy AAA even if the UAS are attritable, as they are vital to the targeting process. This fact is clearly visible in the Case 1 Graphs in Figure 164.

Lastly, like in Research Question 2, both cases clearly proved the importance of the MDTF to reduce delays and expedite data to the HIMARS element.

CHAPTER 7. CONCLUSIONS

7.1 Summary of Overall Research Problem



To support the hypothesis for the overall problem addressed in this thesis, the author's objective was to develop and demonstrate an efficient and holistic framework for a complex system-of-systems to allow for means and ways trades and enable multiple stakeholders to conduct electronic design reviews on an existing system-of-systems in order to analyze technological benefits, limitations and policy impacts for future investment and strategy. To accomplish this task, the author identified three sub-problems that, the author believes, satisfy the overall problem, and sufficiently support the overall hypothesis: how to develop system architecture, ability to evaluate policy and structural changes to the architecture, and ability to determine proper allocations and key elements of the system.

In Section 3.1, the author defined common system engineering terms, introduced the concept of system-of-systems, examined systems thinking, and IPPD (Figure 45). In Section 2.2, Modeling and Simulation in the Military, the author examined the basic concepts of M&S uses in the military, noting the spectrum of analysis (Figure 7) methods available to Army decision-makers spanning from game theory models at highest abstraction levels exercises at the highest end of realism (though absent of ground truth). In the center of this spectrum sat Monte Carlo simulations and wargames are listed just shy of exercises in realism. An overview of the four most common M&S paradigms was reviewed and discussed in an overview regarding Petri Nets in Section 4.5.2, regarding DES in Section 3.3.2, regarding ABM in Section 3.3.3, and regarding SD in Section 3.3.4.

Additionally, the proven method of wargames was examined and compared to modern M&S methods, noting that wargames provide an educational aspect to military leaders in the ability to visualize the effects of their decision, and also by providing an unpredictable opposition force, compared to modern M&S that is mostly used at department-level operations research that, depending on the simulation level (Figure 8: DOD Modeling Hierarchy from [35]) offers little in the way of rapid trades of means and ways due to computing power requirements. Research into the history of DoD M&S found great value in its use, but little appreciable use by operational decision makers as they lack the interactive learning provided by wargames and are black boxes to those outside of operations research. The research into DOD M&S also exposed a significant lack of any discussion of SD as a potential paradigm and found that examination of C4ISR was limited to DES of the computer signals.

Various modeling methodologies were reviewed, namely those of Arnold and Wade (Figure 38: Systems Thinking Systemigram from [67]), Sterman (Figure 48: Sterman's Iterative Modeling Process), Forrester (Figure 47: Forrester's 6-Step SD Modeling Process recreated from [69]) and Ford. In addition to IPPD principles from Schrage and Mavris (Figure 45: Georgia Tech IPPD Approach Flowcharts from [75]). With all of this in mind, the concepts are reflected in the framework that was utilized throughout the document.

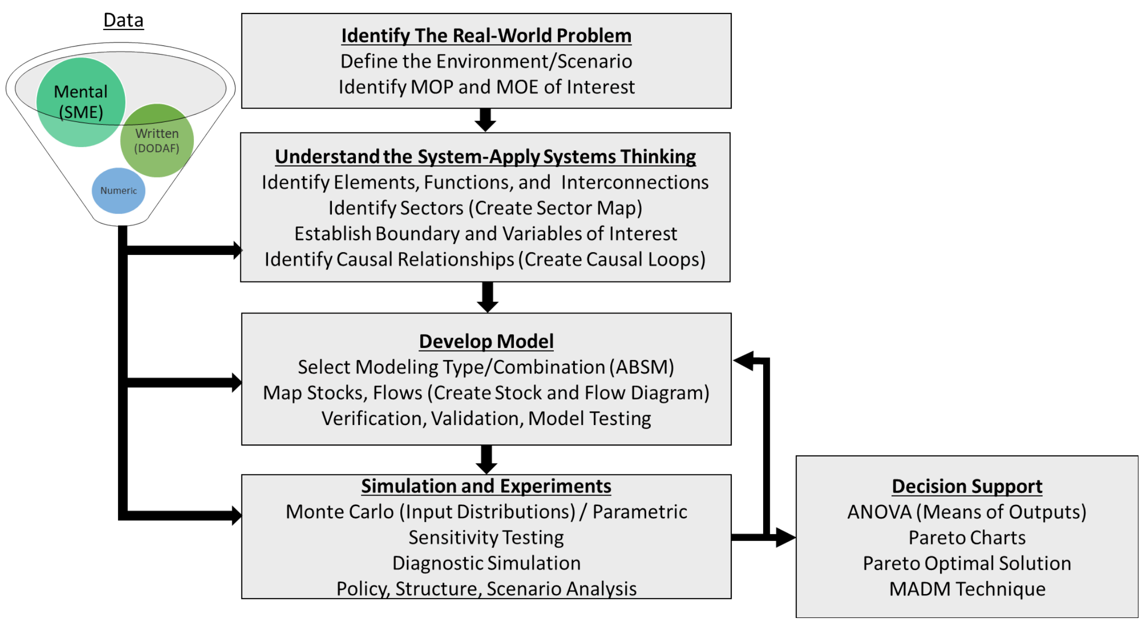


Figure 169: Overall Framework Diagram

The author feels, having, completed the thesis, that the framework developed is suitable for the creation of executable architecture beyond just the AISR use case.

7.1.1 *Problem 1*

Problem one addressed the heart of the problem. Creating executable architecture is an ongoing field of study. However, the author identified that current proposed means to develop executable architecture fell into two categories: manually developed and autogenerated. The manual methods were primarily used to evaluate sections of a system for against design measures of performance such as evaluating data flow to identify choke points. The primary methods used were DES and Petri Nets which can be computational expensive and provide little in the way of interaction and insight like wargames. Alternatively, some researchers have looked at programming autogenerated executable architecture via SysML™. However, such proposals only look at checking completeness and connectivity of the architecture and rely on a single modeling paradigm that may not be suitable for all problems. Furthermore, they require complete DoDAF models to generate. However, lack of standardization within DoDAF itself means that not all documents needed will have been completed nor have many DoDAF for existing SoS even been made in SysML™.

Therefore, a need exists for a holistic method to rapidly develop executable architecture for existing complex SoS to experiment with both means and ways trades with available existing DoDAF products and limited data.

Question 1.1 asked what means were available to aid in the use of systems thinking to understand the system structure as the first step of conceptualizing the model. The author hypothesized that three primary sources of information recommended by Forrester [134] [141] for creating M&S for an executable architecture: the mental database, the written database, and the numeric database were sufficient. The author felt that SME input, and DoDAF models could satisfy the first two requirements. Using the work of previous research as a starting point, DoDAF static architecture viewpoints were mapped to an SD stock and flow diagram (Figure 76: Final Proposed DoDAF Mapping). SME input was added as the mental database to fill in any gaps, constraints, and relationships not identified in the DoDAF. In the absence of data due to classification concerns, parametric ranges were applied to all the variables. Variables were included in the model even if accurate numerical data did not exist or could not be found so as not to overlook any potential influencer on the system behavior.

The author then walked through a step-by-step construction of the SD model for each system and the method by which the final executable architecture was stitched together (Figure 113: Complete ISR PED D3A Model) to include mathematical relationships and major constraints and equations.

Research Question 1.2 sought to demonstrate a method to compare and evaluate different major modeling and simulation paradigms. Each paradigm was described in more detail, reviewing general principles and examples of use. The different paradigms were then be compared against criteria developed from Army modeling and simulation guidance, developed from observations made in CHAPTER 1 regarding desirable traits of executable architectures, regarding stakeholder requirements (Figure 82: Pugh Selection of M&S Paradigm).

The author determined that System Dynamics could provide an overarching M&S architecture that could capture key aspects of the system and enable understanding of technology benefits and limitations, policy impacts, and the likely outcome of future investment strategies as an overarching simulation paradigm. If necessary, the SD model could be supplemented by smaller less aggregated models (ABM, DES). However, having SD serve as the overall wrapper reduced computing requirements significantly.

The author feels the work in Chapter 4 sufficiently addresses the question, supports the selection of SD as the overarching modeling type, and adequately maps the DoDAF to develop an executable architecture with available products through both theoretical mappings and clear demonstration of the mapping through the construction of the executable architecture for the AISR PED in a D3A environment use case.

7.1.2 Problem 2

Research Question 2 hoped to explore the ability of the executable architecture to explore the effects of structural and policy changes on the systems and the systems-of-systems. To show that the SD executable architecture can effectively evaluate such questions, the PED subsystem was utilized to explore specific proposed DOTmLPF-P changes including the addition of the MDTF, technology improvements, and training/experience improvements.

By using a standardized input, the ability to explore input variable and structural changes manually to play games and visualize effects was demonstrated using assumed value ranges. This allowed the interpretation of effects and the user to make more educated assessments as to the impact of variations. The input variables were then assessed over the entire span of ranges using Monte Carlo simulation, sped up by the utilization of Latin Hypercube DOE sampling for 10,000 combination of input variables with near instantaneous results. A method to explore the results via the aid of the initial manual methods was introduced. Finally, once the most influential variables were identified, the ability to optimize the combination given desired outputs was demonstrated using built in Vensim SD functionality.

For the specific MDTF problem, RQ2.2 asked, given new structure, how intelligence and resources should be allocated through the MDTF PED along with additional DOTmLPF-P improvement to improve overall mission effectiveness. Following the method explained above, the executable architecture was able to explore this question given immediate technology and experience values and determine that unequal distribution of observables and manpower is best to increase the output of the PED structure while reducing total overwork of personnel in the three stages. Additional training/experience and technology will moderately increase the total observables process in a given time but will have a greater impact on the reduction of overwork with a constrained number of personnel.

While the EA was successful in this realm, it is necessary to explore the effects given an operational scenario and the complete SoS to compare how these DOTmLPF-P structural and policy changes may be different in the larger context.

7.1.3 Problem 3

Research Question 3 sought to demonstrate the ability of the executable architecture could be used to identify elements and values of the larger SoS architecture that have the greatest impact on operational MOP/MOE. Unlike Research Question 2, the larger SoS had many feedback loops requiring a means to compare trades and observe the effects of multiple subsystems that were built in Section 5.4. While Research Question 2 showed the ability to conduct sensitivity analysis and optimization to find ideal settings for the smaller system, the overall SoS executable architecture had far more variables and the inclusion of stochastic enemy actions that induced highly influential noise.

To account for stochasticity in normally deterministic M&S paradigm, the two cases provide numerically generated inputs based on desirability, but they are not absolutes, nor do they guarantee global maximum. A Latin Hypercube DOE was constructed to include the same 2500 points in the design space for 410 noise variables. The averages of the results for each point were used to construct surrogate models using artificial neural nets. Two cases with different priorities and desirability of MOP/MOE yielded drastically different results to demonstrate the importance of proper MOP/MOE desirability selection. The results were input back into the SD executable architecture as start values to visualize and evaluate the effects over time while making univariate changes to adjust value to realistic or feasible goals. If desired, the overall system could be optimized again in the SD executable architecture by provide narrowed ranges around the values output from the optimization of the surrogate models. This type of optimization was demonstrated in RQ2 and was not repeated in the interest of brevity.

Several key takeaways were observed. First, when placed in the larger operational context against operational MOP/MOE without a standardized input the PED manning, technology, and experience levels varied significantly from the values found in RQ2. Most namely, the allocation of observables and the allocation of MDTF PED personnel. This example highlights the necessity to evaluate SoS holistically in the larger operational context when discerning manning, asset, and technology allocations. The SD executable architecture provides a rapid means to do so with multiple stakeholders.

The use of surrogate models and statistical analysis tools allowed for several different means to identify of the most influential variables and the inputs for the executable architecture. By comparing the very different behaviors for two different sets of recommended inputs base on prioritization of MOP/MOE yields insight into how influential the variables selected are. The variables highlighted in yellow in Table 32 indicate values that changed the most between Case 1 and Case 2, but they do not necessary indicate these variables are the most influential of the influential variables. The variables that did not change between Case 1 and Case 2 are just as important but have more certainty in their values. Thus, when returning to the SD executable architecture those values can be fixed as targetable values and only those highlighted in yellow should be undergo univariate change as necessary to obtain the desired behavior.

All the variables used in the cases were identified in Section 6.3. Interestingly, these variables spanned the DOTmLPF-P spectrum (Figure 168: Important Variables Across DOTmLPF-P Figure 168)

Doctrine				Org.	Training				Materiel						Personnel					
% Obs Direct to MDTF	Average Launch and Transit Time	Total UAS in Theater	% Fires Focused on Enemy Artillery	MDTF Allocation	ACE Fusion Expertise	ED Expertise	MDTF Expertise	PE Expertise	Frequencies Per Satellite	ACE Fusion Technology	ED Technology	MDTF Technology	PE Technology	Satellites Available	PED Personnel Total	ED Personnel	ACE Fusion Personnel	Ground Control Stations Available	Total Crews	Initial UAS

Figure 170: Important Variables Across DOTmLPF-P Sepectrum

Two major exceptions of note were the number of crews and the number of ground control stations. Because of the way that the SD executable architecture is constructed to reflect real limit on real operations, 5 items limit the number of UAS overhead at any given time: Initial UAS (UAS desired overhead); crew limit, ground control station limit, the frequency limit, and the UAS Pool level (see Section 4.6.1). This constraint is not reflected in the surrogate models, as logic cannot be applied. From Table 32, it can be seen that Case 1 and Case 2 call for a significantly different number of Total Crews (30 versus 50) and Ground Control Stations Available (29 versus 50). However, the frequency limit (satellites multiplied by frequencies per satellite) were much lower (15 versus 35) and were thus the limiting factor. Therefore, any number of crews or ground control stations greater than the most restrictive limit has no effect whatsoever on the behavior of the system. If, however, crew or ground control stations were the limiting constraint, they would have set the maximum for the number of UAS Overhead (at least in initially) and had a significant impact on the behavior of the system over time. This observation highlights the importance of having an executable architecture that is both interactive and is built with SMEs and/or stakeholders to improve their knowledge of the system interactions rather than just rely on a data-in, data-out methodology.

This framework also yielded other interesting observations from comparison of the two cases. To obtain maximized enemy to friendly loss ratios, both cases called for maximum UAS in theater (75) but also both called for only 20 Initial UAS. For Case 1, the UAS Overhead were limited to 15 due to the frequency limit, as discussed. In Case 2, however, the frequency limit was 35. The Initial UAS was the limiting value for UAS Overhead. This is significant because it provides a valuable insight that goes against assumed doctrine. To maximize immediate observables and targeting, one could logically expect to put the maximum amount of UAS overhead, but the effect of capping the UAS overhead allowed a longer use of the reserve pool until satellite attrition limited UAS overhead.

Additionally, comparison of the two cases provided insight into proper targeting prioritization with regards to HIMARs focus. In both cases, less than 40% of fires were directed at enemy counter battery, which means most fires were directed at the enemy AAA. This confirms the strategy of first eliminating enemy AAA even if the UAS are attritable, as they are vital to the targeting process. This fact is clearly visible in the Case 1 Graphs in Figure 164.

Lastly, like in Research Question 2, both cases clearly proved the importance of the MDTF to reduce delays and expedite data to the HIMARS element.

The author feels the practical assessments of the use case for the entire SoS in conjunction with the surrogate models to identify the most influential elements of the SoS clearly demonstrates the executable architecture to not only dynamically model system behavior over time, but identify areas for further research. All of this is accomplished in an easy to understand, rapidly developed, interactive simulation paradigm that provides visualization and aids in understanding of the connections, policies, interactions, and their effects, providing a learning element like wargaming that is mission from straign campaign analysis or data driven analytics. ,

7.2 Summary of Contributions

The end goal was to explore and demonstrate a means to develop a rapid, ubiquitous executable architecture that serves as both means to evaluate system-of-system architecture and to easily communicate information to decision-makers (technical orientation and management orientation) using AISR PED architecture as a use case.

This research successfully built upon prior research that explored means by which executable architecture could developed utilizing static DoDAF models. Prior methods either explored Systems Dynamics as a theoretical alternative without creating an executable model or dismissed it entirety based on the false premises that it is solely deterministic or a false equivalency to Discrete Event Simulation. Additionally, previous research constructed simulations as a data-driven output tool within the larger context of a decision-making methodology without consideration of the ability to consider the management orientation and abilities for SMEs to interact with the simulation to observe trends and inform decisions without considerable variation to the models.

This research demonstrates the ability to leverage SD to provide a holistic top-down method and a balance between no simulation and advance simulation for rapid development and assessment. This method is computationally inexpensive and can easily modified to customize to the problem and provide the ability to make trades, generate, repeatable traceable results and add scientific reasoning that current SME generated capabilities assessments often lack. In addition, it allows for the visualization of the second and third order effects and contributions that are otherwise impossible for the human mind to comprehend. It provides a wargame-like interactive experience that can add the managerial element and inform decision makers in a way that is easily communicated and understood and is developed through the combination of SME intuition/experience, existing DoDAF models/products, and limited data.

Beyond executable architecture through simulation, this research demonstrates the benefits of incorporating repeatable stochastic effects into SD models to represent combat environments. Additionally, it demonstrates a means to conduct sensitivity analysis using Monte Carlo simulations that had previously identified as a lacking in static DoDAF models. Additionally, using artificial neural networks, demonstrates the ability to generate surrogate models for instantaneous analysis against operational MOP/MOE.

Expands upon previous body of research of System Dynamics PED analysis previously completed for the Army. This previous research conducted by researchers at Charles River Analytics focused on future PED support systems but did not include the larger operating environment or the experimental Multi-Domain Task Force PED element. Lastly, prior research did not incorporate long range precision fires or enemy activities which are included in this research to provide a more holistic analysis and ability to evaluate operational effectiveness rather than just system performance.

7.3 Future Work

The intent of this thesis was to explore the ability to create a rapidly developed, computationally inexpensive holistic executable architecture for complex systems of system architecture against operational MOP/MOE using existing DoDAF models, SME expertise, and limited data to enable the development and scientific backing of an initial capabilities document during the capabilities-based analysis phase of the Joint Capabilities

Integration and Development System. While this research demonstrated the efficacy of using System Dynamics (namely the commercial platform Vensim) combined with statistical analysis software (JMP) for the AISR PED D3A use case, it relied on global assumptions and imposed stochastic probability models to account for enemy and friendly engagements. While this is suitable for a time-constrained problem set, if time permits, the author feels like the integration of an agent-based simulation paradigm either internally to the SD model or externally and providing data to the SD model would strengthen the global relationships and provide a greater level of realism. Additionally, a Petri Net model or Discrete Event Simulation would aid in the accuracy of the PED to more accurately account for data processing in a less aggregated form. While SD aggregation is sufficient for addressing behavior trends to the overall system-of-systems, it may not be able to account for more nuanced delays and refinement of the data transport and analysis. The ability to rapidly include these features along with the ability to calibrate the model against classified data would prove to be a desirable area of future study.

APPENDIX A. AISR BACKGROUND INFORMATION

A.1 AISR PED D3A Architecture

Legacy Army AISR systems, particularly the Army's manned and unmanned aircraft, provided intelligence directly to an assigned customer based on a tasked collection deck through a process called collection management from organic assets or assigned assets from higher echelons. Depending on the asset, information may have been passed via direct communication with troops on the ground, their tactical operations center (TOC) or via post mission reports sent to the higher tasking authority All-Source Collection Element (ACE) for fusion with other sources of intelligence for use by that command and its subordinates.

Depending on the sensor payloads, the processing of data for exploitation and the production of products was either conducted onboard or post-mission by intelligence analysts belonging to the unit conducting the AISR. With the introduction of full-motion video (FMV) the feed was initially broadcast to the TOC of the supported unit for situation awareness and decision making via tactical common data links (TCDL). Later technological developments allowed the feeds to also be sent to the troops on the ground via One System Remote Video Terminals (OSRVT) and broadcast on the regional computer network using commercial video software. However, the large amounts of video were initially not centrally stored, organized, or catalogued for future use. [126].

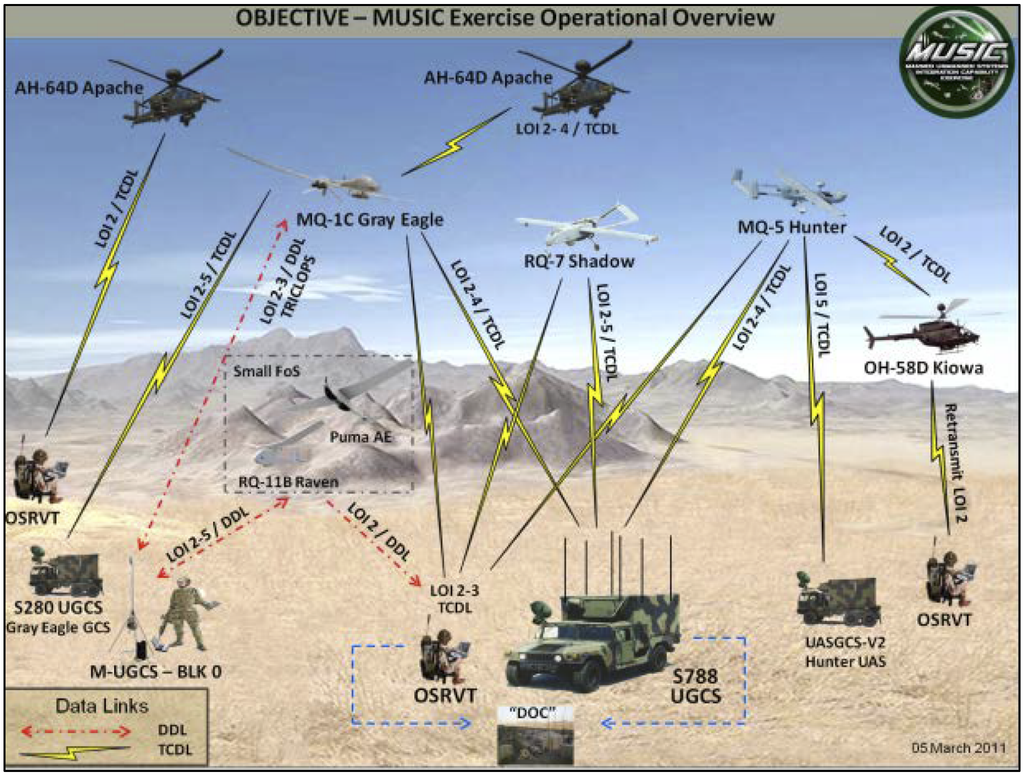


Figure 171: Direct Support UAS Architecture from [126]

This method of operation, while advantageous to the unit directly supported by the assets, limited the distribution of intelligence in a classic stovepipe fashion. Despite assets flying in certain areas from all services, the information collected was not available to other units or services in the area who did not have access to it; there was a significant lack of sharing of resources and information across platforms; some assets suffered from a lack of available analysts to conduct PED while others had too many; and the demand for AISR increased to cover intelligence gaps. The inability to conduct tailored data-sharing meant that even if uploaded to a various systems and databases, analysists were required to sift through an overwhelming amount of data resulting in missed information. Intelligence is only useful if it gets to the right people at the right time. Of course, as demand increased so too did the number of assets, the architecture to support it. As an attempt to improve intelligence, the DOD developed the reach-back architecture, federated PED enterprise, and the distributed common ground station (DCGS) for information sharing.

Though not clearly depicted in the diagram, both manned and unmanned assets can provide sensor data directly to a line-of-site (LOS) gateway or directly to the tactical consumer. However, the only sensor data immediately usable by the tactical consumer is near real-time FMV for situation awareness and rapid decisive decision making on the tactical battlefield. FMV alone, though, is not a complete fused intelligence product.

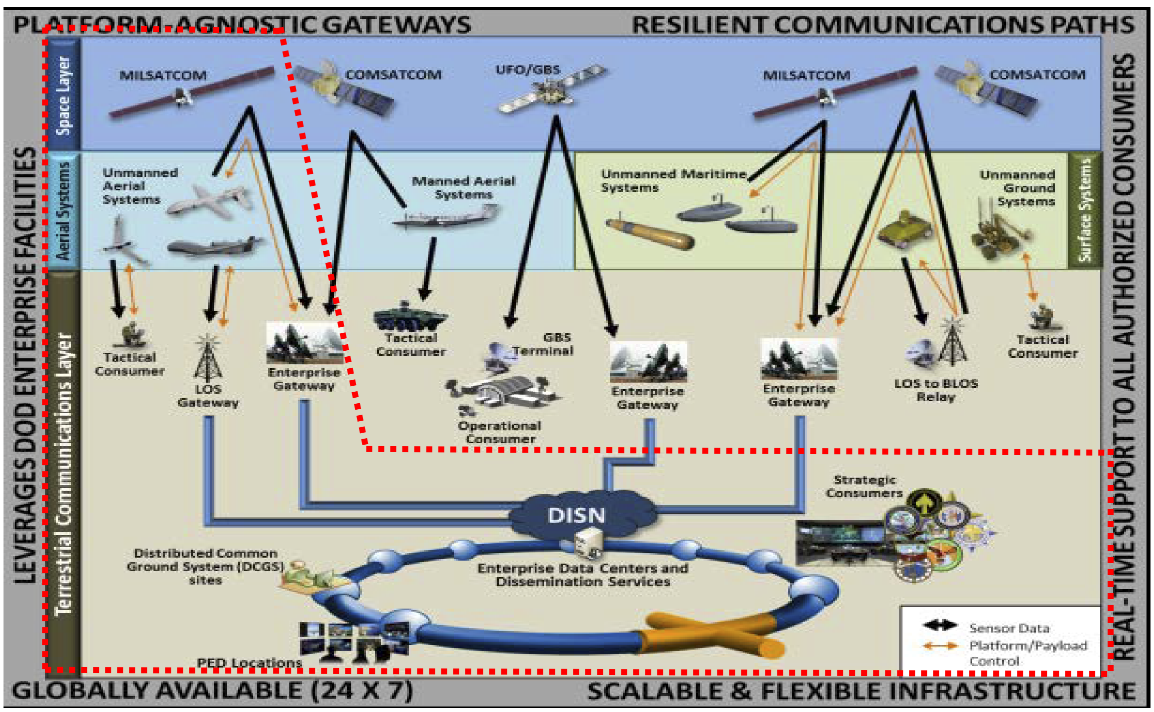


Figure 172: High-Level Command, Control, Communications, and Computers (C4) Operational Overview (OV-1) of ISR System-of-Systems [126]

Once in the enterprise gateway, the raw data is transferred over the Defense Information Systems Network (DISN) transport and Internet Protocol (IP) net-centric services. These systems enable both the BLOS control of UAS and the global distribution of the raw data to Enterprise Data Centers (EDC), Processing Exploitation and Dissemination (PED) sites, and are distributed to the strategic consumers and the distributed common ground system (DCGS) sites at the operationally deployed units. It is important to note that AISR in only part of the data that enters the intelligence enterprise, the others are space systems and surface systems, both ground and maritime, generating a massive amount of data [126]. In the 2013 publicly released information briefing *Army Intelligence 2020 and Beyond*, the Department of the Army's Intelligence Directorate (DA-G2), the slide summarizing the Distributed Common Ground Station-Army (DCGS-A) (Figure 173) depicts this massive amount of accessible data through which intelligence professionals at both the distributed PED sites (depicted on the map) and the operational units must sift, analyze and fuse in a timely manner [153].

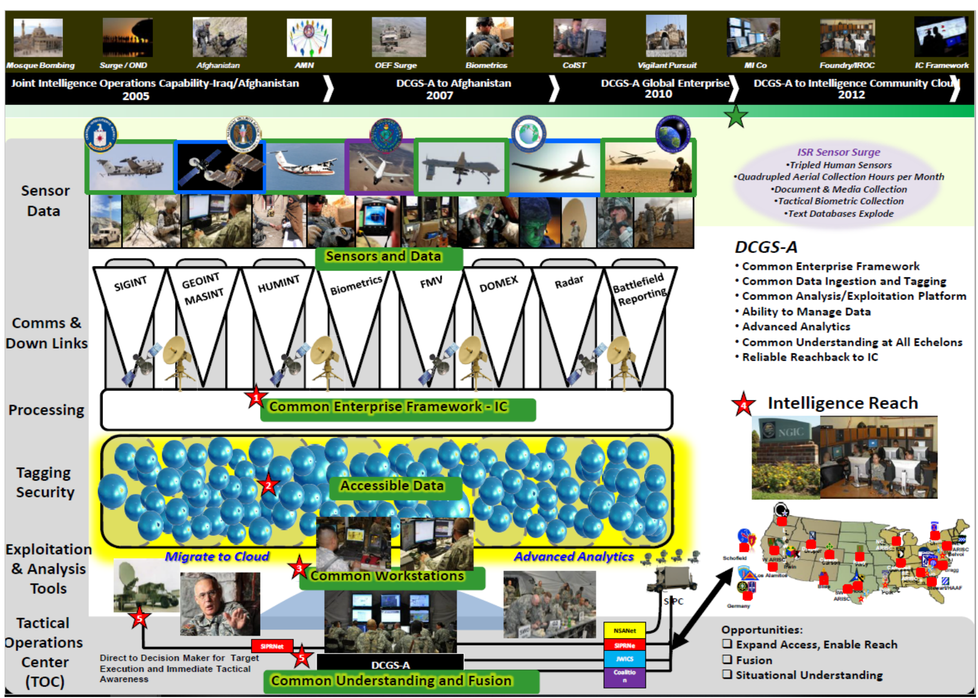


Figure 173: DA-G2 DSGS-A Overview[153]

A.2 Decision Advantage

Over the past 18 years of operations, the used of manned and unmanned AISR has increased exponentially to support urgent Combatant Command (CCMD) requirements throughout the globe [126, 154, 155]. So much so, that the presence of AISR and FMV are virtually assumed. The counterterrorism operations in Afghanistan and Iraq have demonstrated that the AISR platforms:

> have potential strategic and operational impacts requiring near real-time delivery of video and other sensor data to theater operations centers and rear area headquarters to support urgent targeting and force protection decisions [11].

War by nature is complex and dynamic, so too is the current AISR PED distributed architecture. When facing A2AD environments, some argue that our current AISR UAS fleet will become obsolete. For the past 18 years the United States military has had the luxury of air supremacy. In a near-peer environment we will likely have only air parity, or possibly air denial or incapability. Some fear that our current unmanned aircraft are too slow and are thus vulnerable to attack by enemy fighters and air defense arrays. However, it is these same concerns that make it more likely that the United States military will use UAS in environments where the risk to manned aircraft is too high. In fact, the DOD *Unmanned Systems Integration Roadmap 2017-2014* states that "[t]o ensure our military advantage, emphasis should be placed on the evolution, availability, and employment of unmanned technology"[11] The Joint Operational Planning Process describes five phases of a notional strategic war as shown in Figure 177 . They are as follows: Phase 0: Shape, Phase I: Phase II: Seize the Initiative, Phase III: Dominate, Phase IV Stabilize, and Phase V: Enable Civil Authority.[135] AISR will be critical in all of these phases, but will be especially crucial in the Phases 0-1 before hostilities begin as a means to shape future operations and deter threats without leading to unnecessary escalation in the event of a shoot down as was demonstrated in recent war games conducted at MIT.[12] Should hostilities begin, AISR will be crucial to direct long-range precision fires against enemy artillery and air defenses Phase II (Seize the Initiative). to make way for follow-on dominating activities while reducing the friendly casualties.

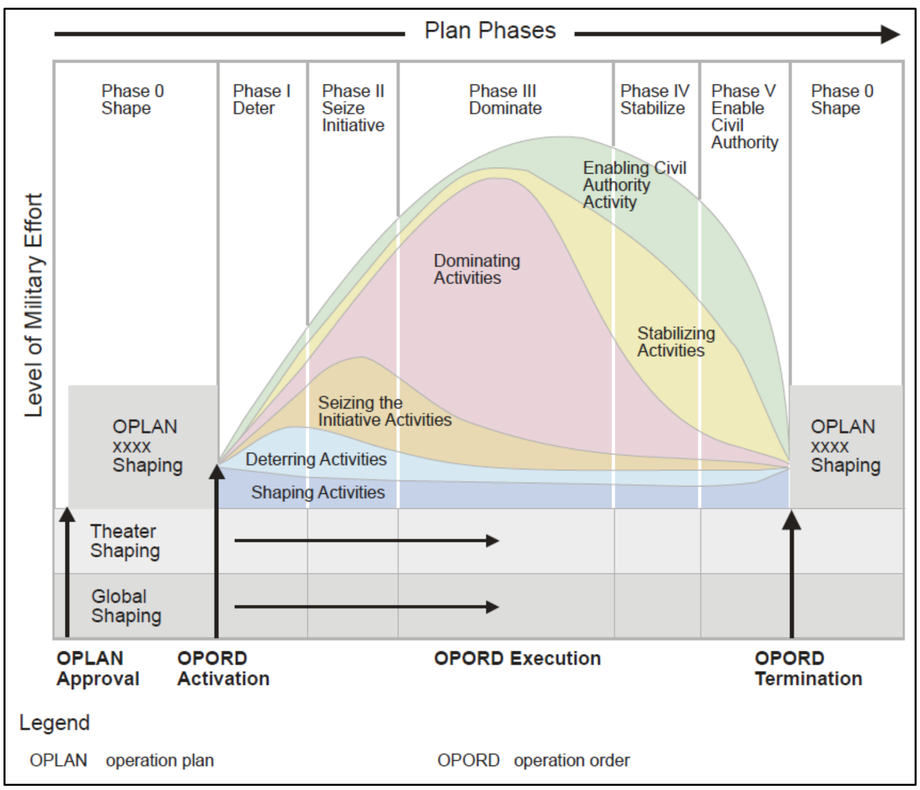


Figure 174: Operational Phases of War from [135]

It is critically important that AISR activities, investments and modernizations efforts be synchronized for economy of force of highly demand but limited assets. As noted in the preceding quotation from the *DOD Unmanned Systems Roadmap 2017-2042*, the requirements of the CCMDs, while primarily for full-motion video (FMV), also required a variety of sensor mixes to include measurement and signals intelligence (MASINT), signals intelligence (SIGINT), and electronic intelligence (ELINT) [149, 156]. These intelligence capabilities and associated analytics have proven critical to provide combatant commanders with a common operating picture needed to make timely decisions and provide a significant advantage on the battlefield.

The 2028 vision for the future of Army AISR lies not in a single collection platform but in a "family of integrated flying systems" deemed the Multi-Domain Sensing System (MDSS) These UAS will be collectively layered at various altitudes and support a variety of consumers at all echelons while providing Multi-INT PED fusion of information for a common operating picture at echelons above brigade. The first priority for the development of MDSS is AISR support to targeting for long-range precision fires (LRPF). This challenge has five elements: sensors; platforms; integration of intelligence, electronic warfare, and cyberspace; data transport, and PED. [124]

For information and data to be used as intelligence and for targeting, the raw data must be processed and exploited before being disseminated. This process can be timely depending on the intelligence type and the desired layering of products to generate a complete intelligence picture. The PED process is what drives the intelligence mission across the Joint enterprise [157]. Delays anywhere in the pathway o include the PED, can result in missed opportunities, failed operations/strikes, and, in the worst case, the loss of friendly lives.

“*Optimal use of ISR now requires much more innovative exploitation skills and accompanying PED technology improvements”*

-MG(R) Eugene Haase

Distributed Common Ground System-Future[158]

Figure 175: Key Takeaway

A.3 Increasing Demand and Data Overload

The prevailing attitude from key decision makers and historic trend data from the last 18 years indicated a tendency towards continuously increasing the number of AISR assets to satisfy the ever-growing requirements for AISR throughout the globe. In fiscal year 2018, the military had an estimated 3,187 UAS in the projected inventory, though this quantity is based solely on budget documents and is not the complete inventory of aircraft but systems (many of which include 3-4 aircraft per system). For the projected FY18 orders the DOD requested an additional 792 UAS, though 94% of those acquisitions were small Group 1 UAS.[159] In 2013, the DOD had over 11,000 UAS aircraft alone. [126]

In the publicly released information briefing *Army Intelligence 2020 and Beyond*, the Department of the Army’s Intelligence Directorate (DA-G2) notes that between 2005 and 2013 the number of aerial collection hours flown per month for the Army’s AISR assets, operated by subordinated units of the United States Army Intelligence and Security Command (INSCOM), quadrupled.[153] This figure doesn’t even include the collection hours flown by Army Shadow (RQ-7) and Gray Eagle (MQ-1C) assets assigned the Army’s eleven active duty Combat Aviation Brigades (CABs) performing reconnaissance, surveillance, and target acquisition (RSTA) roles. Nor does this figure include AISR from the U.S. Air Force, U.S Navy, or the U.S. Marine Corps, which far outnumber Army assets.

In fact, in the year 2016, the Air Force's MQ-1 Predator and the MQ-9 Reaper UAS alone flew a combined 351,000 combat hours and conducted over 40% of kinetic airstrikes.[159] Assuming an equivalent number of FMV collection hours, those two assets collected 40 years' worth of video in a one-year timespan all of which need to be processed, exploited, and disseminated. Obviously, this amount of FMV data alone is unmanageable and is further hindered by lack of tools to correlate and visualize the data, lack of cross-cueing, inadequate sharing, and the inability to aggregate data and reports. Currently, intelligence analysts at supported commands must 'pull' information from data which is typically segregated by type of source, service, and region instead of having relevant reports and data "pushed" to them.[160] This challenge has led to the call for new technologies such as big data analytics, machine learning and artificial intelligence to be infused. To address this issue the DOD has also established the Joint Artificial Intelligence Center (JIAC) and the Army has established the Army Artificial Intelligence Task Force (AAI-TF).[161] Additionally, the Army has proposed the MDTF introduced in section 0 as an alternative structure to redirect targeting specific information to maneuver units and LRPF elements before the LTIOV.

A.4 Major Monetary Investment

According to the Center for the Study of the Drone at Bard College's annual reports: *Drones in the Defense Budget, Navigating the Fiscal Year 2018 Budget Request* and *Summary of Drone Spending in the FY19 Defense Budget Request*, the DOD requested

$6.97 billion in the 2018 budget for all autonomous systems (commonly referred to in civilian terms as "drones") related expenditures and $9.39 billion in 2019—a 26% increase. AISR, particularly in the terms of UAS, is a major portion of that expense for the United States DOD. In 2018, $5 billion (71%) of the $6.97 billion was budgeted for UAS. Of that $5 billion, $4.1 billion was allocated toward operational systems; namely the RQ-11 Raven, RQ-4 Global Hawk, and the MQ-8 Fire Scout. In 2019 $6.05 billion (64%) of the $9.39 billion was budgeted for UAS. Therefore, while total UAS expenditure has increased by $1 billion, the portion of the "drone" budget has decreased. This decreasing percentage of the overall drone budget has been the trend since 2016 as major procurements peaked in 2014. However, though not forecasted in the 2013 UAS roadmap, the Army and the Air Force did request an additional 16 new MQ-9 Reapers for the Air Force and an additional 11 new MQ-1C Gray Eagles for the Army in 2018. In 2019, the Air Force has requested an additional 29 MQ-9 Reapers and 120 new RQ-20 Pumas while the Army has requested 10 new MQ-1C Gray Eagles, 600 new RQ-11 Ravens, 10 new RQ-7 Shadows and 1084 new Soldier Borne Sensors (SBS), totaling approximately 3,070; in terms of numbers, it is the largest procurement in six years. Of note, the SBS will be a short-range micro UAS and is still in development. It is envisioned to be a lightweight aircraft, potentially commercial off the shelf (COTS) that can be carried by individual members of infantry squads and easily deployed for quick-look reconnaissance at a lower cost than the RQ-11 Ravens which cost $180,400 each. [159, 162] Like the RQ-11 Raven the SBS is for immediate use at the tactical level and will not be connected to the reach-back architecture into the larger federated PED enterprise.

It is important to understand that the expenses discussed refer to funds needed to develop, test, procure, field, operate and maintain the UAS and their associated support equipment. These numbers do not include the development of the PED Architecture, the analytical framework to conduct the analysis, or the investment in AI to address the data overload. To date, the Army has spent over $3 billion dollars on DCGS-A which is necessary for intelligence professionals to gather and analyze all the information. But the information is too overwhelming, the system is too cumbersome, and it requires a steep learning curve to be effective. In March 2019, the Army announced it would grant commercial company Palantir an $800 million dollar contract to replace the custom-built platform in an effort to improve the end user ability to organize and make use of all of the intelligence data.[163] Lastly, in the fiscal year 2019 budget, the Pentagon has increased funding for artificial intelligence program, Project Maven, by 81% to over $100 million.[162]

APPENDIX B. VENSIM EQUATIONS AND UNITS

B.1 Primary AISR PED D3A Architecture

B.1.1 Global Parameters

FINAL TIME = 48
Units: Hour

NOISE SEED=3
Units: dmnl
Range: [0,500,1]

TIME STEP = 0.0625
Units: Hour
Range: [0,?]

B.1.2 Friendly UAS Equations

Average Launch and Transit Time=3
Units: Hour
Range: [2,6,0.5]

Controllable UAS per Frequency= 1
Units: UAS/frequency
Range: [1,5,1]

Crew Limit= Crews Available*UAS Per Crew
Units: UAS [0,?,1]

Crews Available=20
Units: Crew [0,50,1]

Detection Rate= UAS Overhead*Observable Output(Target Saturation Ratio)
Units: observables/Hour

Enemy Satellite Attack= RANDOM POISSON (0, Satellites Overhead, 0.002*Satellites Overhead, 0 , 1 ,0)/TIME STEP
Units: sat/Hour

Enemy Anti Aircraft Fire= IF THEN ELSE (UAS Overhead>0:AND: Enemy Anti Aircraft >= 1, RANDOM POISSON(0, UAS Overhead, 0.0125*UAS Overhead*Enemy AAA Ratio , 0 ,1 , 0)/TIME STEP,0)

Units: UAS/Hour

Frequencies Per Satellite=2

Units: frequency/sat

Range: [1,10,1]

Frequency Limit= Controllable UAS per Frequency*Number of Frequencies Available

Units: UAS

Ground Control Station Limit= Ground Control Station UAS Capacity*Ground Control Stations Available

Units: UAS

Ground Control Station UAS Capacity=1

Units: UAS/GCS

Range: [1,5,1]

Ground Control Stations Available=20

Units: GCS

Range: [0,50,1]

Initial UAS=30

Units: UAS

Range: [1,50,1]

Network Load Capacity=0.75

Units: dmnl

Range: [0,1,0.1]

Number of Frequencies Available= Frequencies Per Satellite*Satellites Overhead

Units: frequency

Range: [0,?]

Observable Output ([(0,0)-(200,5)],(0,0),(1,1),(2,2),(3,3),(4,4),(5,5),(200,5))

Units: observables/(Hour*UAS)

OR Rate=0.8

Units: dmnl

Range: [0,1,0.05]

Satellites Available= 10

Units: sat

Range: [1,15,1]

Satellites Overhead= INTEG (-Enemy Satellite Attack, Satellites Available)
Units: sat

Target Saturation Ratio=INTEGER(ZIDZ(INTEGER(Enemy Anti Aircraft+Enemy Artillery),INTEGER(UAS Overhead))*UAS per Target)
Units: dmnl

Total Observables Rate=Detection Rate*Network Load Capacity
Units: observables/Hour

Total UAS in Theater=50
Units: UAS
Range: [0,75,1]

UAS Capacity Loss= IF THEN ELSE (UAS Limit<=UAS Overhead, PULSE (Time, TIME STEP)*(UAS Overhead -UAS Limit)/TIME STEP, 0)
Units: UAS/Hour

UAS Limit=(MIN((MIN(Crew Limit, Ground Control Station Limit)), MIN(Frequency Limit, Initial UAS)))
Units: UAS

UAS Overhead= INTEG (UAS Replacement Rate-Enemy Anti Aircraft Fire-UAS Capacity Loss, UAS Limit)
Units: UAS

UAS Per Crew= 1
Units: UAS/Crew
Range: [1,4,1]

UAS per Target=1
Units: UAS/targets [1,1,1]

UAS Pool= INTEG (-UAS Replacement Rate, IF THEN ELSE(Total UAS in Theater*OR Rate-UAS Limit<=0, 0 , Total UAS in Theater *OR Rate-UAS Limit))
Units: UAS

UAS Replacement Delay= DELAY FIXED (Enemy Anti Aircraft Fire, Average Launch and Transit Time, 0)
Units: UAS/Hour

UAS Replacement Rate=IF THEN ELSE(UAS Overhead<UAS Limit :AND: UAS Pool>0 , MIN(UAS Replacement Delay , UAS Pool/TIME STEP) , 0)
Units: UAS/Hour

B.1.3 Friendly Fires Equations

"% Fires Focused on Enemy AAA"= 1-"% of Fires Focused on Enemy Artillery"
Units: dmnl

"% of Fires Focused on Enemy Artillery"= 0.5
Units: dmnl
Range: [0,1,0.1]

AAA Targeting Rate= IF THEN ELSE(Enemy Anti Aircraft>0:AND:Targets>=1 , MIN((RANDOM BINOMIAL(0, MIN(Targets , Max Engagements) , Fire Accuracy, MIN(Targets , Max Engagements) , 0, "% Fires Focused on Enemy AAA" , 0)/TIME STEP),"% Fires Focused on Enemy AAA" *Targets/TIME STEP) , 0)
Units: targets/Hour

Artillery Targeting Rate= IF THEN ELSE (Enemy Artillery>0: AND: Targets>=1, MIN ((RANDOM BINOMIAL (0, MIN (Targets, Max Engagements) , Fire Accuracy, MIN (Targets, Max Engagements), 0, "% of Fires Focused on Enemy Artillery" , 0)/TIME STEP),"% of Fires Focused on Enemy Artillery"*Targets/TIME STEP) , 0)
Units: targets/Hour

Fire Accuracy= Hit Probability*Kill Probability
Units: dmnl
Range: [0,1,0.02]

Hit Probability= 0.5
Units: dmnl
Range: [0,1,0.01]

Initial Number of HIMARS=20
Units: system
Range: [0,40,1]

Kill Probability=0.5
Units: dmnl

Max Engagements= Max Rate of Fire*Number of HIMARS
Units: targets

Max Rate of Fire=4
Units: targets/system
Range: [1,6,1]

Number of HIMARS= INTEG (-Enemy Counter Battery Fire, Initial Number of HIMARS)
Units: system

Range: [0,?]

B.1.4 Fusion Equations

"% Targets Expired per Hour"= 0.2
Units: dmnl
Range: [0,1,0.1]

ACE Fusion Capacity Utilization= Capacity Utilization Graph (ACE Fusion Ratio)
Units: 1/Hour

ACE Fusion Expertise=3
Units: products/(personnel)
Range: [0,10,0.5]
ACE Fusion Limit=ACE Fusion Expertise*ACE Fusion Personnel*ACE Fusion Technology
Units: products

ACE Fusion Overwork= SMOOTH (IF THEN ELSE (ACE Fusion Capacity Utilization>=Overwork Threshold, ACE Fusion Capacity Utilization-Overwork Threshold, 0), time to average)
Units: 1/Hour

ACE Fusion Personnel=20
Units: personnel
Range: [1,30,1]

ACE Fusion Ratio=ZIDZ (Disseminated Intel Backlog, ACE Fusion Limit)
Units: 1

ACE Fusion Technology=1
Units: dmnl
Range: [1,3,0.1]

AVG Observables per Product=3
Units: observables/products

AVG Product per Target=3
Units: products/targets

Capacity Utilization Graph=([(0,0)-(6,2)],(0,0),(0.2,0.4),(0.4,0.7),(0.6,0.85),(0.8,0.88),(1,0.9),(1.2,0.92),(1.4,0.94),(1.6,0.96),(1.8,0.98),(2,1),(6,1))
Units: 1/Hour

DI Overflow Rate=Disseminated Intel Backlog*Overflow Graph(ACE Fusion Ratio)

Units: products/Hour

Disseminated Intel Backlog= INTEG (((ED/AVG Observables per Product)-DI Overflow Rate-Fusion Rate),0)
Units: products [0,?]

Disseminated Intel Overflow= INTEG (DI Overflow Rate,0)
Units: products

ED= ED Limit*ED Capacity Utilization
Units: observables/Hour

Fusion Rate= ACE Fusion Capacity Utilization*ACE Fusion Limit
Units: products/Hour

MDTF PEF=MDTF Capacity Utilization*MDTF PE Limit
Units: observables/Hour

Overflow Graph=([(0,0)-(10,1)],(0,0),(10,1))
Units: 1/Hour

Overwork Threshold=0.85
Units: 1/Hours
Range: [0.75,1,0.5]

Target Overflow= INTEG (Target Overflow Rate,0)
Units: targets

Target Overflow Rate=IF THEN ELSE(AAA Targeting Rate=0 :AND: Artillery Targeting Rate =0 :OR: Targets> Max Engagements , PULSE TRAIN(Time , TIME STEP , 1 , FINAL TIME)*INTEGER("% Targets Expired per Hour" *Targets)/TIME STEP , 0)
Units: targets/Hour

Targets= INTEG ((((Fusion Rate+MDTF PEF)/AVG Product per Target)-AAA Targeting Rate-Artillery Targeting Rate-Target Overflow Rate) 0)
Units: targets
Range: [0,?]

time to average= 1
Units: Hour
Range: [0.5,2,0.5]

B.1.5 Enemy Assets Equations

AAA Destruction Rate= MIN(AAA Targeting Rate, Enemy Anti Aircraft/TIME STEP)

Units: targets/Hour

Ability to Replace=0
Units: dmnl
Range: [0,1,0.05]

Artillery Destruction Rate= MIN (Artillery Targeting Rate, Enemy Artillery/TIME STEP)
Units: targets/Hour

Enemy AAA Ratio= ZIDZ(Enemy Anti Aircraft,Initial Enemy AAA)
Units: 1

Enemy Anti Aircraft= INTEG (Rate of Replacement AAA-AAA Destruction Rate, Initial Enemy AAA)
Units: targets

Enemy Artillery= INTEG (Rate of Replacement Artillery-Artillery Destruction Rate, Initial Enemy Artillery)
Units: targets
Range: [0,100,1]

Enemy Artillery Ratio=ZIDZ (Enemy Artillery, Initial Enemy Artillery)
Units: 1

Enemy Counter Battery Fire=IF THEN ELSE (Enemy Artillery>=1:AND: Number of HIMARS> 0, RANDOM POISSON (0, Number of HIMARS, 0.007*Number of HIMARS*Enemy Artillery Ratio , 0 , 1, 0)/TIME STEP ,0)
Units: system/Hour

Enemy Delay Interval=3
Units: Hour
Range: [0,10,0.5]

Enemy Destroyed= INTEG (AAA Destruction Rate + Artillery Destruction Rate,0)
Units: targets

Enemy Replacement Delay AAA=DELAY FIXED (AAA Destruction Rate, Enemy Delay Interval, 0)
Units: targets/Hour

Enemy Replacement Delay Artillery= DELAY FIXED (Artillery Destruction Rate, Enemy Delay Interval, 0)
Units: targets/Hour

Initial Enemy AAA=75
Units: targets
Range: [0,100,1]

Initial Enemy Artillery=75
Units: targets
Range: [0,100,1]

Rate of Replacement AAA= Ability to Replace*Enemy Replacement Delay AAA
Units: targets/Hour

Rate of Replacement Artillery= Enemy Replacement Delay Artillery*Ability to Replace
Units: targets/Hour

B.2 PED Sub-Model Architecture

B.2.1 Processing Exploitation

PE=PE Capacity Utilization*PE Limit
Units: observables/Hour

PE Avg Cap Utilization=SMOOTH(PE Capacity Utilization, time to average)
Units: 1/Hour

PE Capacity Utilization=Capacity Utilization Graph (PE Observables Ratio)
Units: 1/Hour

PE Expertise=3
Units: observables/(personnel) [0,10,0.5]

PE Limit=PE Personnel*PE Expertise*PE Technology
Units: observables

PE Observables= INTEG (PE-ED-ED Overflow,0)
Units: observables

PE Observables Ratio=ZIDZ(Observables Backlog, PE Limit)
Units: dmnl

PE Overflow=Observables Backlog*Overflow Graph(PE Observables Ratio)
Units: observables/Hour

PE Overflow Ratio=ZIDZ (PE Overflow Total, Observables Total)
Units: 1

PE Overflow Total= INTEG (PE Overflow,0)
Units: observables

PE Overwork=SMOOTH(IF THEN ELSE(PE Capacity Utilization>=Overwork Threshold , PE Capacity Utilization-Overwork Threshold , 0), time to average)
Units: 1/Hour

PE Personnel=PED Personnel Total-MDTF PEF Personnel
Units: personnel
Range: [0,50,1]

PE Technology=1
Units: dmnl
Range: [1,3,0.1]

PED Observables=Total Observables Rate*(1-"% Obs Direct to MDTF")
Units: observables/Hour

PED Personnel Total=25
Units: personnel
Range: [0,50,1]

B.2.2 Exploitation Dissemination

ED=ED Limit*ED Capacity Utilization
Units: observables/Hour

ED Avg Cap Utilization=SMOOTH(ED Capacity Utilization, time to average)
Units: 1/Hour

ED Capacity Utilization=Capacity Utilization Graph(ED Observables Ratio)
Units: 1/Hour

ED Expertise=3.5
Units: observables/(personnel)
Range: [0,10,0.5]

ED Limit=ED Expertise*ED Personnel*ED Technology
Units: observables

ED Observables= INTEG (ED,0)
Units: observables

ED Observables Ratio=ZIDZ(PE Observables, ED Limit)
Units: dmnl

ED Overflow=PE Observables*Overflow Graph(ED Observables Ratio)
Units: observables/Hour

ED Overflow Ratio=ZIDZ(ED Overflow Total, Observables Total)
Units: 1

ED Overflow Total= INTEG (ED Overflow,0)
Units: observables

ED Overwork=SMOOTH(IF THEN ELSE(ED Capacity Utilization>=Overwork Threshold , ED Capacity Utilization-Overwork Threshold , 0), time to average)
Units: 1/Hour

ED Personnel=10
Units: personnel
Range: [1,30,1]

ED Technology=1
Units: dmnl
Range: [1,3,0.1]

B.2.3 Multi-Domain Task Force

"% Obs Direct to MDTF"=0.3
Units: dmnl
Range: [0,1,0.01]

MDTF Allocation=0.25
Units: dmnl [0,1,0.01]

MDTF Avg Cap Utilization=SMOOTH(MDTF Capacity Utilization, time to average)
Units: 1/Hour

MDTF Capacity Utilization=Capacity Utilization Graph(MDTF PEF Obs Ratio)
Units: 1/Hour

MDTF Observable Backlog= INTEG (MDTF Priority Observables-MDTF Overflow-MDTF PEF,0)
Units: observables

MDTF Overflow=MDTF Observable Backlog*Overflow Graph(MDTF PEF Obs Ratio)

Units: observables/Hour

MDTF Overflow Ratio=ZIDZ(MDTF Overflow Total,Obervables Total)
Units: 1

MDTF Overflow Total= INTEG (MDTF Overflow,0)
Units: observables

MDTF Overwork=SMOOTH(IF THEN ELSE(MDTF Capacity Utilization>=Overwork Threshold , MDTF Capacity Utilization -Overwork Threshold , 0), time to average)
Units: 1/Hour

MDTF PE Limit=MDTF PEF Expertise*MDTF PEF Personnel*MDTF PEF Technology
Units: observables

MDTF PEF=MDTF Capacity Utilization*MDTF PE Limit
Units: observables/Hour

MDTF PEF Expertise=2
Units: observables/personnel

MDTF PEF Obs Ratio=ZIDZ(MDTF Observable Backlog,MDTF PE Limit)
Units: dmnl

MDTF PEF Personnel=PED Personnel Total*MDTF Allocation
Units: personnel [0,50,1]

MDTF PEF Technology=1
Units: dmnl

MDTF Priority Observables="% Obs Direct to MDTF"*Total Observables Rate
Units: observables/Hour

Priority Targets= INTEG (MDTF PEF,0)
Units: observables

B.2.4 General

Capacity Utilization Graph([(0,0)-(6,2)],(0,0),(0.2,0.4),(0.4,0.7),(0.6,0.85),(0.8,0.88),(1,0.9),(1.2,0.92),(1.4,0.94),(1.6,0.96),(1.8,0.98),(2,1),(6,1))
Units: 1/Hour

Observables Total= INTEG (Observables Copy,0)

Units: observables

Observables Backlog= INTEG (PED Observables-PE-PE Overflow,0)
Units: observables

Observables Copy=Total Observables Rate
Units: observables/Hour

Overflow Graph([(0,0)-(10,1)],(0,0),(10,1))
Units: 1/Hour

Overwork Threshold=0.85
Units: 1/Hour [0.75,1,0.5]

time to average=1
Units: Hour [0.5,2,0.5]

Total Observables Rate=Detection Rate*Network Load Capacity
Units: observables/Hour

Total Overflow Ratio=ZIDZ((ED Overflow Total+PE Overflow Total+MDTF Overflow Total),Observables Total)
Units: 1

B.3 Measures of Interest Output Direct Calculations

AAA to UAS Loss Ratio=XIDZ(Enemy AAA Losses,Friendly UAS Losses,Enemy AAA Losses/Unit UAS)
Units: targets/UAS

Artillery Loss Ratio=XIDZ(Enemy Artillery Losses,HIMARS Losses,Enemy Artillery Losses/Unit Artillery)
Units: targets/system

Cumulative Overflow= INTEG (DI Overflow Rate+MDTF Overflow+ED Overflow+PE Overflow,0)
Units: observables

Cumulative Overwork= INTEG (ACE Fusion Overwork+ED Overwork+MDTF Overwork+PE Overwork,0)
Units: 1

Enemy AAA Percent Loss=ZIDZ(Enemy AAA Losses,Initial Enemy AAA)*100
Units: 1

Enemy Artillery Losses= INTEG (Artillery Destruction Rate,0)
Units: targets

Enemy Artillery Percent Loss=ZIDZ(Enemy Artillery Losses, Initial Enemy Artillery)*100
Units: dmnl

Friendly UAS Losses= INTEG (Enemy Anti Aircraft Fire,0)
Units: UAS

HIMARS Losses= INTEG (Enemy Counter Battery Fire,0)
Units: system

HIMARS Percent Loss=ZIDZ(HIMARS Losses,Initial Number of HIMARS)*100
Units: dmnl

Satellite Percent Loss=ZIDZ(Satellite Losses, Satellites Available)*100
Units: dmnl

Time to Eliminate Enemy AAA=IF THEN ELSE(Enemy Anti Aircraft=0, 1 , 0)
Units: 1

Time to Eliminate Enemy Artillery=IF THEN ELSE(Enemy Artillery=0, 1 , 0)
Units: 1

Total Loss Ratio=XIDZ((Enemy AAA Losses+Enemy Artillery Losses),(HIMARS Losses+Friendly UAS Losses),((Enemy AAA Losses+Enemy Artillery Losses)Unit friendly))
Units: enemy/friendly

UAS Loss Percent=ZIDZ(Friendly UAS Losses,Total UAS in Theater)*100
Units: dmnl

REFERENCES

[1] (2017). *Modernization Priorities for the United States Army*. Available: https://vtol.org/files/dmfile/10062017_Army_Modernization1.pdf

[2] Z. Alessi-Friedlander, "Learning to Win While Fighting Outnumbered: General Donn A. Starry and the Challenge of Institutional Leadership during a Period of Reform and Modernization," *Military Review,* 2017.

[3] (2015). *OSD001694-15/CMD001984-15, Wargaming and Innovation*. Available: https://dod.defense.gov/news/publications-archive/

[4] J. Judson, "The Army is Creating a Modernization Command to Keep Projects on Track," in *Defense News* vol. 2019, ed. Online, 2017.

[5] *The U.S. Army in Multi-Domain Operations 2028* (TRADOC Pam 525-3-1). Fort Eustis, VA: U.S. Army Training and Doctrine Command, 2018.

[6] M. Peck, "The Return of Wargaming: How DoD Aims to Re-Imagine Warfare," in *GovTech Works* vol. 2019, ed: General Dynamics Information Technology, 2016.

[7] United States Senate. 115, 1 Session. (2017). *Posture Of The Department Of The Army*. Available: https://www.armed-services.senate.gov/imo/media/doc/17-54_05-25-17.pdf

[8] C. Pellerin, "Deputy Secretary: Third Offset Strategy Bolsters America's Military Deterrence," in *DoD News* vol. 2019, ed. Washington, D.C.: Defense Media Activity, 2016.

[9] N. Hollenbeck and B. Jensen. (2017, 21December 2018). *Why the Army Needs a Futures Command*. Available: https://warontherocks.com/2017/12/army-needs-futures-command/

[10] in *Merriam-Webster*, ed. Online, 2019.

[11] (2017). *Unmanned Systems Integrated Roadmap 2017-2042*. Available: https://news.usni.org/2018/08/30/pentagon-unmanned-systems-integrated-roadmap-2017-2042

[12] E. Lin-Greenberg, "Game of Drones: The Effect of Remote Warfighting Technology on Conflict Escalation (Evidence from Wargames)," *SSRN Electronic Journal,* 5 September 2018.

[13] "Army Futures Command – Task 5 NPS Student / Faculty Team Overview Brief v2.0," Presentation, December 2018.

[14] D. L. Suits. (April 24, 2019). *What Soldiers Need to Know About Multi-Domain Ops, Modernization* [Online]. Available: https://www.army.mil/article/220663/what_soldiers_need_to_know_about_multi_domain_ops_modernization

[15] *Department of Defense Dictionary of Military and Associated Terms* (JP 1-02). Department of Defense, 2013.

[16] O. o. A. Studies, *Capabilities-Based Assessment (CBA) Handbook: A Practical Guide to the Capabilities-Based Assessment*. 2014, p. 91.

[17] (28 April 2019). *DOTMLPF*. Available: https://ssl.armywarcollege.edu/dde/documents/jsps/terms/DOTMLPF.cfm

[18] R. M. G. Fogg. (2019, 20 Dec 2019). *From the Big Five to Cross Functional Teams: Integrating Sustainment into Modernization*. Available: https://www.tradoc.army.mil/Publications-and-Resources/Article-Display/Article/1984287/from-the-big-five-to-cross-functional-teams-integrating-sustainment-into-modern/

[19] J. Gould, "McHugh: Army Acquisitions 'Tale of Failure'," in *Defense News* vol. 2019, ed, 2015.

[20] in *Business Dictionary*, ed: WebFinance Inc., 2019.

[21] G. J. Roedler, Jones, Cheryl, "Technical Measurement," INCOSE-TP-2003-020-01, 27 December 2005 2005, Available: http://www.psmsc.com/Downloads/TechnologyPapers/TechnicalMeasurementGuide_v1.0.pdf.

[22] P. Wills, "Joint Capabilities Integration and Development System (JCIDS) A Primer," 22 Feb 2017, 2017.

[23] Chairman of the Joint Chiefs of Staff. (2015). *CJCSI 3170.01I, Chairman of the Joint Chiefs of Staff Instruction Joint Capabilities Integration and Development System (JCIDS)*. Available: https://www.jcs.mil/Library/

[24] (2018). *The Army Vision*. Available: https://www.army.mil/e2/downloads/rv7/vision/the_army_vision.pdf?_st

[25] M. Ambrose, "Defense Acquisition System Overview," ed: Defense Acquisition University 2018.

[26] (2008). *Modeling And Simulation Guidance For The Acquisition Workforce*. Available: http://www.acqnotes.com/Attachments/Modeling%20&%20Simulation%20Guidance%20for%20the%20Acquisition%20Workforce.pdf

[27] (2018). *Manual for the Operation of the Joint Capabilities Integration and Development System*. Available: https://www.dau.mil/cop/rqmt/DAU%20Sponsored%20Documents/Manual%20-%20JCIDS,%2031%20Aug%202018.pdf

[28] S. H. Dam, *DoD Architecture Framework 2.0 – A Guide to Applying System Engineering to Develop Integrated, Executable Architectures*. Manassas, VA: SPEC Innovations, 2014.

[29] C. Dickerson and D. N. Mavris, *Architecture and Principles of Systems Engineering* (Complex and Enterprise Systems Engineering Model). Auerbach Publications, 2009.

[30] A. H. Levis and L. W. Wagenhals, "C4ISR architectures: I. Developing a process for C4ISR architecture design," *Systems Engineering,* vol. 3, no. 4, pp. 225-247, 2000.

[31] (2019). *DOD Architecture Framework Version 2.02*. Available: https://dodcio.defense.gov/Library/DoD-Architecture-Framework/

[32] (2020, 10 July). *SysML Open Source Project - What is SysML? Who created SysML?* Available: https://sysml.org/

[33] R. Karban, "“Using Executable SysML Models to Generate Systems Engineering Products"," in *NoMagic World Symposium*, Allen, TX, 2016.

[34] W. J. Hanoka and M. H. Ryan, "A Study of Executable Model Based Systems Engineering from DODAF Using Simulink," M.S. Dissertation, Systems and Engineering Management, DEPARTMENT OF THE AIR FORCE AIR UNIVERSITY, Wright-Patterson Air Force Base, Ohio, AFIT/GSE/ENV/12-S05DL, 2012.

[35] R. R. Hill and J. O. Miller, "A History of United States Military Simulation," presented at the Proceedings of the 2017 Winter Simulation Conference, Las Vegas, Nevada, 2017.

[36] R. R. Hill, J. O. Miller, and G. A. McIntyre, "Simulation Analysis: Applications of Discrete Event Simulation Modeling To Military Problems," presented at the Proceedings of the 33nd Conference on Winter Simulation, Arlington, Virginia, 2001.

[37] Joint Chiefs of Staff. (2009). *Capabilities-Based Assessment (CBA) User’s Guide*. Available: http://acqnotes.com/wp-content/uploads/2014/09/Capabilities-Based-Assessment-CBA-Users-Guide-version-3.pdf

[38] S. Bonder, "Army Operations Research—Historical Perspectives and Lessons Learned," *Operations Research,* vol. 50, no. 1, pp. 25-34, 2002.

[39] *Joint Planning* (JP 5-0). Washington D.C., USA: Joint Publishing Directorate, 2017.

[40] *Military Decisionmaking Process* (Lessons and Best Practices). Fort Leavenworth, KS: United States Army Combined Arms Center, 2015.

[41] *Commander and Staff Organization and Operations* (FM 6-0). Washington D.C., USA: Department of the Army, 2014.

[42] B. Wade, "Critical Elements of Analytic Wargame Design," *Phalanx: The Magazine of National Security Analysis,* vol. 51, no. 4, p. 5, December 2018 2018.

[43] M. Caffrey, "Toward a History-Based Doctrine for Wargaming," *Air and Space Power Journal,* vol. 14, no. 3, p. 24, Apr 27, 2000 2000.

[44] J. Markley, *Strategic Wargaming Series Handbook*. Carlsisle, PA: United States Army War College Strategic Wargaming Division, 2015.

[45] H. J. Brightman and M. K. Dewey, "Trends in Modern War Gaming: The Art of Conversation," *Naval War College Review,* vol. 67, no. 1, p. 15, Winter 2014 2014.

[46] T. E. Kleynhans and P. H. Spies, "The influence of Newtonian-Cartesian scientific thinking on the interpretation of welfare and development," *Development Southern Africa,* vol. 11, no. 1, pp. 49-67, 1994/02/01 1994.

[47] (4 April 2019). *Defense M&S Catalog*. Available: https://www.msco.mil/DoDTools/DoDEnterpriseManagementTools/MSCatalog.aspx

[48] J. Morrison, D. Goldsmith, and M. Siegel, *Dynamic Complexity in Military Planning: A Role for System Dynamics*. 2008.

[49] J. Z. Dave Panson, Don Emerson, "AFSIM Campaign Simulation Approach Overview," 2019.

[50] T. S. Pugsley, "An Implicit Model Development Process for Bounding External, Seemingly Intangible/Non-Quantifiable Factors," PhD Dissertation, Operations Research, Naval Post Graduate School, Monterey, California, 2017.

[51] J. Sterman, *Business Dynamics: Systems Thinking and Modeling for a Complex World*. Boston, MA: McGraw Hill, 1999.

[52] D. V. Holt and M. Osman, "Approaches to Cognitive Modeling in Dynamic Systems Control," (in English), *Frontiers in Psychology,* Perspective vol. 8, no. 2032, 2017-November-29 2017.

[53] J. Sterman *et al.*, "WORLD CLIMATE:A Role-Play Simulation of Climate Negotiations," *Simulation & Gaming,* vol. 46, no. 3-4, pp. 348-382, 2015.

[54] J. Garcia, "Executable Architecture Analysis Modeling," in *SpringSim '07*, Norfolk, Virginia, 2007, vol. 2, pp. 177-184: Society for Computer Simulation International.

[55] S. Mittal, "Extending DoDAF to Allow Integrated DEVS-Based Modeling and Simulation," *Special issue: DoDAF, Journal of Defense Modeling and Simulation*

JDMS, vol. III, no. 2, 2006.

[56] R. Wang and C. H. Dagli, *An Executable System Architecture Approach to Discrete Events System Modeling Using SysML in Conjunction with Colored Petri Net*. 2008, pp. 1-8.

[57] K. Griendling and D. N. Mavris, "Development of a dodaf-based executable architecting approach to analyze system-of-systems alternatives," in *2011 Aerospace Conference*, 2011, pp. 1-15.

[58] K. A. Griendling, "Architect: The Architecture-Based Technology Evaluation and Capability Tradeoff Method," PhD Dissertation, Aerospace Engineering, Georgia Institute of Technology, Atlanta, GA USA, 2011.

[59] A. Maheshwari, C. Kenley, and D. DeLaurentis, "Creating Executable Agent-Based Models Using SysML," *INCOSE International Symposium,* vol. 25, pp. 1263-1277, 10/01 2015.

[60] B. Committee on Making the Soldier Decisive on Future, S. Board on Army, Technology, E. Division on, S. Physical, and C. National Research, *Making the Soldier Decisive on Future Battlefields*. Washington, D.C., UNITED STATES: National Academies Press, 2013.

[61] J. D. W. Morecroft, *Strategic Modelling and Business Dynamics + Website : A Feedback Systems Approach*. New York: John Wiley & Sons, Incorporated, 2015.

[62] (May 2015). *Army Training Publication 3-60 (FM 3-60): Targeting*.

[63] (2011). *Modeling and Simulation (M&S) Glossary*. Available: www.msco.mil

[64] (2008). *Systems Engineering Guide for Systems of Systems*. Available: https://www.acq.osd.mil/se/docs/SE-Guide-for-SoS.pdf

[65] J. Banks, "Discerete Event Simulation," presented at the Winter Simulation Conference, 1999.

[66] M. W. Maier, "Architecting principles for systems-of-systems," *Systems Engineering,* vol. 1, no. 4, pp. 267-284, 1998.

[67] R. D. Arnold and J. P. Wade, "A Definition of Systems Thinking: A Systems Approach," *Procedia Computer Science,* vol. 44, pp. 669-678, 2015/01/01/ 2015.

[68] D. Cabrera, "System Dynamics ≠ Systems Thinking," in *System Thinking Daily* vol. 2019, ed. Ithaca, NY: Cabrera Research Lab, 2015, p. Online.

[69] J. W. Forrester, "System Dynamics, Systems Thinking, and Soft OR," vol. 10, no. 2, pp. 245-256, 1994.

[70] P. M. Senge, *The fifth discipline : the art and practice of the learning organization*, Rev. and updated.. ed. New York: New York : Doubleday/Currency, 2006.

[71] B. Richmond, "Systems thinking/system dynamics: Let's just get on with it," *System Dynamics Review,* vol. 10, no. 2-3, pp. 135-157, 1994.

[72] D. V. Behl and S. Ferreira, "Systems Thinking: An Analysis of Key Factors and Relationships," *Procedia Computer Science,* vol. 36, pp. 104-109, 2014/01/01/ 2014.

[73] C. Lofdahl, "Applying System Dynamics to Military Operations," presented at the Proceedings of the 5th international conference on Social Computing, Behavioral-Cultural Modeling and Prediction, College Park, MD, 2012. Available: https://link.springer.com/chapter/10.1007%2F978-3-642-29047-3_33

[74] J. Morecroft and S. Robinson, "Comparing discrete-event simulation and system dynamics: modelling a fishery," in *Proceedings of the 2006 OR Society Simulation Workshop*, 2006, pp. 137-148: Ashorne Hill Conference Centre, England.

[75] D. P. Schrage and D. Mavris, "Technology for affordability - How to define, measure, evaluate, and implement it?," in *American Helicopter Society 50th Annual Forum*, Washington, DC, 1994: American Helicopter Society.

[76] O. Balci, "Validation, verification, and testing techniques throughout the life cycle of a simulation study," *Annals of Operations Research,* journal article vol. 53, no. 1, pp. 121-173, December 01 1994.

[77] G. B. Sv. (2004, May 5,). *Modeling & Simulation: An Introduction*. Available: http://www.systems-thinking.org/modsim/modsim.htm

[78] J. Banks, *Discrete-event system simulation*, 2nd ed.. ed. Upper Saddle River, N.J.: Upper Saddle River, N.J. : Prentice Hall, 1996.

[79] C. A. Chung, *Simulation Modeling Handbook: A Practical Approach*, 1st Edition ed. Boca Raton, FL: CRC Press, 2003.

[80] "Verification, Validation, and Accreditation (VV&A) of Federations," Research and Technology OrganizationTR-MSG-019, April 2008, Available: https://apps.dtic.mil/dtic/tr/fulltext/u2/a485631.pdf.

[81] S. Robinson, *Simulation Verification, Validation and Confidence: A Tutorial*. 1999, pp. 63-69.

[82] S. Pezeshk. (2014, 5 January). *Aleatory Variability and Epistemic Uncertainty*. Available: http://www.ce.memphis.edu/7137/PDFs/Abrahamson/C05.pdf

[83] M. Silva, "50 years after the PhD thesis of Carl Adam Petri: A perspective," *IFAC Proceedings Volumes,* vol. 45, no. 29, pp. 13-20, 2012/01/01/ 2012.

[84] V. Volovoi, "Modeling of system reliability Petri nets with aging tokens," *Reliability Engineering & System Safety,* vol. 84, no. 2, pp. 149-161, 2004/05/01/ 2004.

[85] R. Melnyk, "Petri Nets: An Alternative to Markov Chains," *The Journal of the Reliability Analysis Center,* no. First Quarter, 2004, 2004.

[86] K. Griendling and D. Mavris, "An architecture-based approach to identifying system-of-systems alternatives," in *2010 5th International Conference on System of Systems Engineering*, 2010, pp. 1-6.

[87] A. Levis, "A Colored Petri Net Model of Intellegent Nodes," presented at the IMACS Symposium on Modeling and Control of Technological Systems, Lille, France, 05/05, 1991.

[88] (2007). *DoD Architecture Framework Version 1.5*. Available: https://dodcio.defense.gov/Portals/0/Documents/DODAF/DoDAF_Volume_II.pdf

[89] *DOD Architecture Framework Version 2.02, Change 1 Architect's Guide*. Washington, D.C.: Department of Defense, 2015, p. 231.

[90] A. Borshchev and A. Filippov, "From System Dynamics and Discrete Event to Practical Agent Based Modeling: Reasons, Techniques, Tools," presented at the The 22nd International Conference of the System Dynamics Society, Oxford, England, July 25 - 29, 2004.

[91] N. Matloff, *Introduction to discrete-event simulation and the simpy language*. 2008.

[92] A. A. Tako and S. Robinson, *Model building in System Dynamics and Discrete-event Simulation: a quantitative comparison*. 2008.

[93] J. S. Morgan, S. Howick, and V. Belton, "A toolkit of designs for mixing Discrete Event Simulation and System Dynamics," *European Journal of Operational Research,* vol. 257, no. 3, pp. 907-918, 2017.

[94] A. A. Tako and S. Robinson, "Comparing discrete-event simulation and system dynamics: users' perceptions," *Journal of the Operational Research Society,* vol. 60, no. 3, pp. 296-312, 2009/03/01 2009.

[95] S. Bangsow, *Use cases of discrete event simulation appliance and research*. New York: Springer, 2012.

[96] I. S. Ugur Z. Yildirim, Barbaros Tansel, "A Simulation Model for Military Deployment," presented at the Proceedings of the 39th conference on Winter simulation: 40 years! The Best is Yet to Come, Washington D.C., 2007. Available: https://dl.acm.org/citation.cfm?id=1351782

[97] A. T. Bellocchio, "A framework to enable rotorcraft maintenance free operating periods," Ph.D. Dissertation, Aerospace Engineering, Georgia Institute of Technology, Atlanta, GA, USA, 2018.

[98] E. Saltmarsh, "A Modeling Trade-Off Forecasting Environment for Military Aircraft Sustainment," Ph.D. Dissertation, Aerospace Engineering, Georgia Institute of Technology, Atlanta, GA USA, 2015.

[99] E. M. Ekman, "Simulating sustainment for an Unmanned Logistics System concept of operation in support of distributed operations," Masters, Operations Research, Naval Postgraduate School, Monteray, CA, 2017.

[100] C. Swinerd and K. R. McNaught, "Design classes for hybrid simulations involving agent-based and system dynamics models," *Simulation Modelling Practice and Theory,* vol. 25, pp. 118-133, 2012/06/01/ 2012.

[101] E. Bonabeau, "Agent-based modeling: Methods and techniques for simulating human systems," *Proceedings of the National Academy of Sciences,* vol. 99, no. suppl 3, pp. 7280-7287, 2002.

[102] M. Garifullin, A. Borshchev, and T. Popkov, "USING ANYLOGIC AND AGENT-BASED APPROACH TO MODEL CONSUMER MARKET," in *EUROSIM 2007*, Ljubljana, Slovenia, 2007.

[103] N. Schieritz and P. M. Milling, "Modeling the Forest or Modeling the Trees A Comparison of System Dynamics and Agent-Based Simulation," presented at the 21st International Conference of the System Dynamics Society : July 20 - 24, New York City, USA, 2003.

[104] W. W. Wakeland, E. J. Gallaher, L. M. Macovsky, and C. A. Aktipis, "A Comparison of System Dynamics and Agent-Based Simulation Applied to the Study of Cellular Receptor Dynamics," in *37th Annual Hawaii International Conference on System Sciences, 2004. Proceedings of the*, 2004, p. 10 pp.

[105] H. Scholl, *Agent-Based and System Dynamics Modeling: A Call for Cross Study and Joint Research*. 2001, p. 8 pp.

[106] T. M. Cioppa, T. W. Lucas, and S. M. Sanchez, "Military applications of agent-based simulations," in *Proceedings of the 2004 Winter Simulation Conference, 2004.*, 2004, vol. 1, pp. 1-180.

[107] L. E. Champagne and R. R. Hill, "A Simulation Validation Method Based on Bootstrapping Applied to an Agent-based Simulation of the Bay of Biscay

Historical Scenario," *The Journal of Defense Modeling and Simulation,* vol. 6, no. 4, pp. 201-212, 2009.

[108] J. W. Forrester, *Industrial dynamics* (no. xv, 464 p.). [Cambridge, Mass.]: M.I.T. Press, 1962, pp. xv, 464 p.

[109] J. W. Forrester. (1995) The beginning of system dynamics. *McKinsey Quarterly*. Available: https://www.mckinsey.com/business-functions/strategy-and-corporate-finance/our-insights/the-beginning-of-system-dynamics

[110] (2018, 6 January). *System Dynamics Society*. Available: https://www.systemdynamics.org

[111] (2019). *Electronics Tutorials*. Available: https://www.electronics-tutorials.ws/systems/closed-loop-system.html

[112] D. D. Boettner, *Thermal-Fluid Systems*. West Point, NY: United States Military Academy, 2010.

[113] J. D. S. M. Anjali Sastry, "Desert Island Dynamics: An Annotated Survey of the Essential Systems Dynamics Literature," in *International System Dynamics Conference*, Cancun, Mexico: Monterrey Institute of Technology, 1993, pp. 466-475.

[114] J. W. Forrester, "14 "obvious truths"," *System Dynamics Review,* vol. 3, no. 2, p. 4, Summer 1987 1987.

[115] J. W. Forrester, "System Dynamics and the Lessons of 35 Years," in *The Systemic Basis of Policy Making in the 1990s*, K. B. D. Greene, Ed. Cambridge: Sloan School of Management Massachusetts Institute of Technology, 1991.

[116] S. V. Victor Tang, "System Dynamics: Origins, development, and future prospects of a method," 2001.

[117] T. Rebs, M. Brandenburg, and S. Seuring, "System Dynamics Modeling For Sustainable Supply Chain Management: A Literature Review And Systems Thinking Approach," *Journal of Cleaner Production,* vol. 208, pp. 1265-1280, 2019/01/20 2019.

[118] S. Alessi and B. Kopainsky, "System Dynamics and Simulation/Gaming:Overview," *Simulation & Gaming,* vol. 46, no. 3-4, pp. 223-229, 2015.

[119] J. Wang, "A System Dynamics Simulation Model for a Four-Rank Military Workforce," DEFENCE SCIENCE AND TECHNOLOGY ORGANISATION EDINBURGH (AUSTRALIA) LAND OPERATIONS DIVAug 2007 2007, Available: https://apps.dtic.mil/dtic/tr/fulltext/u2/a474508.pdf.

[120] C.-Y. Fan, P.-S. Fan, and P.-C. Chang, "A system dynamics modeling approach for a military weapon maintenance supply system," *International Journal of Production Economics,* vol. 128, no. 2, pp. 457-469, 2010/12/01/ 2010.

[121] G. V. Amico, "The Application of System Dynamics Techniques to War Game Modeling," Masters, University of Central Florida, Orlando, FL, 125669, 1973.

[122] M. Voshell *et al.*, "Application of Emerging Technologies to Support Army Processing Exploitation and Dissemination (TAPED)," Charles River Analytics, Cambridge, MA2013.

[123] C. Lofdahl, M. Voshell, and S. Mahoney, "Designing Future Processing, Exploitation, and Dissemination Support Systems Using Simulation," *Procedia Computer Science,* vol. 36, pp. 33-40, 2014.

[124] W. Adams, M. Dotson, J. McAfee, F. Ziemba, and D. DuQuesnay, "Fixing Echelons above Brigade Sensor Challenges in Multi-Domain Operations," *MI Professional Bulletin,* vol. 45, no. 3, p. 7, July-September 2019 2019.

[125] N. J. Gulosh, "Employment of Intelligence, Surveillance, and Reconnaissance Drone Swarms To Enhance Ground Combat Operations," M.S. Dissertation, Operations Research, Naval Post Graduate School, Monterey, California, 2018.

[126] (2013). *14-S-0553, Unmanned Systems Integrated Roadmap FY2013-2038*. Available: https://apps.dtic.mil/docs/citations/ADA592015

[127] (2012). *DoDD 3000.09, Department of Defense Directive 3000.09: Autonomy in Weapon Systems*. Available: https://www.hsdl.org/?abstract&did=726163

[128] S. Kimmons. (2018, 21 May). *Multi-Domain Task Force set to lead Pacific Pathways rotation in first overseas tests*. Available: https://www.army.mil/article/207072/multi_domain_task_force_set_to_lead_pacific_pathways_rotation_in_first_overseas_tests

[129] C. Lee. (2019) NEWS FROM AUSA GLOBAL: Army Sharpens Focus on Multi-Domain Warfare. *National Defense*. Available: http://www.nationaldefensemagazine.org/articles/2019/3/27/army-integrates-multi-domain-task-force-unit

[130] A. Bueno, L. Torres Carreño, D. Delgado, and R. Llamosa-Villalba, *Executable Architecture based on System Dynamics: An Integrated Methodology Composed by Standard System Dynamics Modelling and DoDAF Operational View Models*. 2014.

[131] (May 5, 2008). *DoDAF Product Development Questionnaire Analysis Report and New Product Recommendations Report Version 4*.

[132] B. Bagdatli, "Architecture-Based Selection of Modeling Type For System Of Systems Analysis," PhD Dissertation, Aerospace Engineering, Georgia Institute of Technology, Atlanta, GA, 2018.

[133] M. Wayson. (2010). *DoDAF V2.0 Community Update Overview*. Available: https://dodcio.defense.gov/Portals/0/Documents/DODAF/DoDAF_V2-0_Architectural_Views_Examples.ppt

[134] (2020, 24 April). *Catia No Magic Users Guide*. Available: https://docs.nomagic.com/display/NMDOC/No+Magic+Documentation

[135] *Joint Operations*, Change 1 ed. (JP 3-0). Washington D.C., 2017.

[136] H. Rahmandad and J. Sterman, "System Dynamics or Agent-Based Models? Wrong question! Seek the right level of aggregation.," *System Dynamics Society,* Accessed on: 13 May 2019Available: https://www.systemdynamics.org/assets/docs/sdorabm.pdf

[137] L. Lättilä, P. Hilletofth, and B. Lin, "Hybrid simulation models – When, Why, How?," *Expert Systems with Applications,* vol. 37, no. 12, pp. 7969-7975, 2010/12/01/ 2010.

[138] Zharova A.V., "The Practical Methods Of Integration Of System Dynamics And Agent-Based Modeling," *GISAP: Economics, Jurisprudence and Management,* vol. 1, 2013.

[139] S. Burge, "Pugh Matrix," The Systems Engineering Tool BoxRugby Warwickshire, U.K.: Burge Houghes Walsh, 2009. [Online]. Available: https://www.burgehugheswalsh.co.uk/uploaded/1/documents/pugh-matrix-v1.1.pdf.

[140] T. Pawlowsk, P. C. Barr, and S. J. Ring, "Applying Executable Architectures to Support Dynamic Analysis of C2 Systems," presented at the 2004 Command and Control Research and Technology Symposium: The Power of Information Age Concepts and Technologies, 2004. Available: https://apps.dtic.mil/sti/pdfs/ADA466014.pdf

[141] L. Breierova and M. Choudhari, "An Introduction to Sensitivity Analysis," M. I. o. Technology, Ed., ed. Cambridge, MA: MIT System Dynamics in Education Project, 2001.

[142] (2020). *Vensim User Guide*. Available: https://www.vensim.com/documentation/index.html?users_guide.htm

[143] A. Ford, *Modeling the Environment*, Second Edition ed. Washington, DC: Island Press, 2010.

[144] G. E. P. Box, "Robustness in the Strategy of Scientific Model Building," in *Robustness in Statistics*, R. L. Launer and G. N. Wilkinson, Eds.: Academic Press, 1979, p. 202.

[145] D. S. Alberts and R. E. Hayes, *Code of Best Practice: Experimentation* (Information Age Transformation Series). Washington D.C.: Department of Defense, 2009.

[146] Y. Barlas, "Formal aspects of model validity and validation in system dynamics," *System Dynamics Review,* vol. 12, no. 3, p. 28, Fall 1996 1996.

[147] (2012). *ADP 2-0 Intelligence*.

[148] (2013). *FM 3-55 Information Collection*.

[149] *Intelligence Officer's Handbook* (FM 34-8-2). Washington, D.C.: Department of the Army, 1998.

[150] D. P. Kroese, T. Brereton, T. Taimre, and Z. I. Botev, "Why the Monte Carlo method is so important today," *WIREs Computational Statistics,* vol. 6, no. 6, pp. 386-392, 2014.

[151] J. M. Fürbringer and C. A. Roulet, "Comparison and combination of factorial and Monte-Carlo design in sensitivity analysis," *Building and Environment,* vol. 30, no. 4, pp. 505-519, 1995/10/01/ 1995.

[152] K. Hinkelmann, *Design and Analysis of Experiments. Volume 3, Special Designs and Applications*. Hoboken, NJ: Hoboken, NJ : Wiley-Interscience, 2012.

[153] (2013). *Army Intelligence 2020 and Beyond*. Available: www.dami.army.pentagon.mil/

[154] A. Stone, "How Full-Motion Video is Changing ISR," in *C4ISRNET*, ed. Online: Sightline Media Group, 2016.

[155] L. Greenemeier, "The Drone Wars: 9/11-Inspired Combat Leans Heavily on Robot Aircraft," in *Scientific American* vol. 2019, ed. Online: SCIENTIFIC AMERICAN, a Division of Springer Nature America, Inc., 2011.

[156] R. S. H. III, "Air Force Manned Reconnaissance at a Crossroads," in *War on the Rocks*, ed, 2019.

[157] J. J. A. Jr. (2015, 15 May 2019). *Army, Air Force Joint Interoperability Provides Intelligence Products for Consumers*. Available: https://www.army.mil/article/141855/army_air_force_joint_interoperability_provides_intelligence_products_for_consumers

[158] E. Haase, "Distributed Common Ground System–Future: Moving into the 22nd Century Today," *Joint Force Quarterly,* vol. 77, no. 2nd Quarter, p. 5, April 2015 2015.

[159] D. Gettinger, "Drones in the Defense Budget: Navigating the Fiscal Year 2018 Budget Request," Center for the Study of the Drone at Bard College, Annandale-on-Hudson, NYOctober 24, 2017, Available: https://dronecenter.bard.edu/report-drones-in-the-2018-defense-budget/.

[160] J. A. Kimminau, "Five Examples of Big Data Analytics and the Future of ISR," *Joint Force Quarterly,* vol. 77, no. 2nd Quarter, p. 2, April 2015 2015.

[161] B. Wade, "Artificial Intelligence, Machine Learning, and Data Science," Presentation, 2019.

[162] D. Gettinger, "Summary Of Drone Spending In The Fy 2019 Defense Budget Request," Center for the Study of the Drone at Bard College, Annandale-on-Hudson, NYApril 2018, Available: https://dronecenter.bard.edu/report-drones-in-the-2018-defense-budget/.

[163] S. Harris, "Palantir wins competition to build Army intelligence system," in *The Washington Post*, ed. Online, 2019.